



TRABALLO FIN DE GRAO
GRAO EN ENXEÑARÍA INFORMÁTICA
MENCIÓN EN TECNOLOXÍAS DA INFORMACIÓN

Aplicación móbil para a realización e seguimento de rutinas en nenos/as con TEA

Estudante: Noé Vila Muñoz
Dirección: Paula María Castro Castro
Óscar Fresnedo Arias

A Coruña, setembro de 2020.

Á miña familia e amigos

Agradecementos

Á miña familia porque sen ela este proxecto non sería posible.

Aos meus amigos por apoiarme estivese no punto que estivese.

Aos meus titores por guiarme e axudarme en todo momento durante este proxecto.

Resumo

O obxectivo deste traballo fin de grao é o desenvolvemento dunha aplicación móbil multiplataforma para os sistemas operativos Android e iOS que indique e axude a nenos e nenas con algún tipo de trastorno do espectro autista coa realización de rutinas e tarefas cotiás outorgándolles unha maior autonomía e facilitando aos seus educadores e educadoras unha programación e un seguimento destas rutinas e tarefas.

Abstract

The objective of this final degree project is the development of a multiplatform mobile application for Android and iOS operating systems that indicates and helps kids with some type of autism spectrum disorder to carry out daily routines and tasks, providing them greater autonomy and facilitating their educators the programming and monitoring of these routines and tasks.

Palabras chave:

- Trastorno do espectro autista
- Aplicacións móbiles
- Metodoloxías áxiles
- Flutter
- Educación

Keywords:

- Autism spectrum disorder
- Mobile app
- Agile methodologies
- Flutter
- Education

Índice Xeral

1	Introdución	1
1.1	Trastorno de Espectro Autista (TEA)	1
1.2	Motivación	2
1.3	Obxectivos	2
1.4	Estrutura	3
2	Trastorno do Espectro Autista	5
2.1	Revisión histórica	6
2.2	Visión xeral sobre o TEA	6
2.2.1	Cando comeza un TEA?	7
2.2.2	Cando se pode diagnosticar un TEA?	7
2.2.3	Que trazos característicos teñen as persoas con TEA?	7
2.2.4	Que se inclúe na categoría dos TEA?	8
3	Estado da Arte	11
3.1	Día a Día	11
3.2	In-TIC Agenda	11
3.3	Diario de Autismo	11
3.4	Choiceworks	12
3.5	Pictorario	12
3.6	Táboa comparativa	12
4	Fundamentos e Ferramentas Tecnolóxicas	15
4.1	Colección de pictogramas de ARASAAC	15
4.2	Intellij IDEA	16
4.3	Android Studio	16
4.4	Xcode	16
4.5	Dart	17

4.6	Flutter	17
4.7	Firebase	17
4.7.1	Cloud Firestore	18
4.7.2	Google Cloud Storage	18
4.7.3	Firebase Authentication	18
4.8	Android	18
4.9	iOS	19
4.10	LATEX	19
4.11	Git	19
4.12	Microsoft Project 2016	19
4.13	Visual Paradigm Online Diagrams	20
4.14	Balsamiq Wireframes	20
5	Metodoloxía	21
5.1	Definición	21
5.2	Roles	22
5.2.1	Product Owner	22
5.2.2	SCRUM Master	22
5.2.3	Development Team	22
5.3	Sprints	22
5.3.1	Reunión de planificación e revisión	23
5.3.2	Análise	23
5.3.3	Diseño	23
5.3.4	Implementación	23
5.3.5	Probas e solución de erros	23
5.4	Artefactos	24
5.4.1	Product Backlog	24
5.4.2	Sprint Backlog	24
5.4.3	Incremento	24
6	Planificación, seguimento e custos	27
6.1	Recursos humanos	27
6.1.1	Product Owner	27
6.1.2	SCRUM Master	28
6.1.3	Analista	28
6.1.4	Deseñador	28
6.1.5	Programador	28
6.2	Recursos materiais / software	28

6.2.1	Recursos materiais	28
6.2.2	Recursos software	29
6.3	Planificación inicial	29
6.4	Custos	32
6.5	Seguimento	33
7	Análise de Requisitos	35
7.1	Actores	35
7.1.1	Usuario non autenticado	35
7.1.2	Usuario autenticado	35
7.1.3	Administrador	35
7.2	Requisitos	36
7.2.1	Requisitos funcionais (administrador)	36
7.2.2	Requisitos funcionais (usuario)	37
7.2.3	Requisitos non funcionais	38
7.3	Casos de uso	38
7.3.1	Casos de uso do usuario non autenticado	39
7.3.2	Casos de uso do administrador	39
7.3.3	Casos de uso para o usuario	42
7.4	Fluxo da aplicación	44
7.5	Esquema de Datos da Aplicación	44
8	Deseño	47
8.1	Patrón de deseño	47
8.2	Modelo lóxico da base de datos	48
8.2.1	Colección Administradores	48
8.2.2	Colección Usuarios	48
8.2.3	Subcolección Tareas	48
8.2.4	Subcolección Rutinas	50
8.2.5	Subcolección Tareas de Rutinas	50
8.2.6	Subcolección Estadísticas	50
8.3	Deseño da interface de usuario	51
8.3.1	Material Design	51
8.3.2	Bosquexos iniciais de deseño da interface de usuario	52
8.4	Idioma da aplicación	57
8.5	Estrutura do proxecto	58

9	Implementación	61
9.1	Widgets para as vistas	61
9.1.1	Composición de widgets	61
9.2	Implementación dos controladores para os diferentes módulos de Firebase . .	61
9.3	API ARASAAC	64
9.4	Utilizar imaxes propias do dispositivo móbil	65
9.5	Implementación das rutinas	67
9.5.1	Cambios respecto ao deseño inicial	67
9.5.2	Formato do campo duración das tarefas	68
9.5.3	Formato dos momentos (fin e comezo) de asignación das tarefas dentro dunha rutina	68
9.5.4	Compatibilidade das tarefas dentro da rutina	68
9.6	Implementación das estatísticas	69
9.6.1	Tempo utilizado	69
9.6.2	Tarefas ben feitas	70
9.6.3	Tarefas mal feitas	70
9.6.4	Estatísticas dentro da base de datos	70
9.7	Programación asíncrona	71
9.7.1	Refresco da pantalla do neno	72
10	Probas	73
10.1	Probas funcionais	73
10.1.1	Probas unitarias	73
10.1.2	Probas de integración	74
10.1.3	Probas de aceptación	74
10.2	Probas non funcionais	75
10.2.1	Probas de compatibilidade	75
10.2.2	Probas de usabilidade	76
11	Conclusións	77
11.1	Conclusións	77
11.2	Comparativa	78
11.3	Liñas futuras	78
A	Acta de reunión inicial	83
B	Acta de reunión de seguimento	85
C	Acta de reunión final	87

Relación de Acrónimos	91
Bibliografía	93

Índice de Figuras

4.1	Exemplo de pictogramas da colección de ARASAAC.	16
5.1	Representación gráfica sinxela de todos os compoñentes da metodoloxía SCRUM. Imaxe creada a partir da imaxe [1] baixo unha licencia Creative Commons 4.0 que permite a súa edición e uso.	25
6.1	Diagrama de Gantt correspondente á planificación.	31
6.2	Tarefas, duración, custos e recursos asignados detallados.	32
7.1	Diagrama de casos de uso correspondente ao usuario non autenticado.	39
7.2	Diagrama de casos de uso correspondente ao menú do administrador.	41
7.3	Diagrama de casos de uso correspondente á xestión de usuarios do adminis- trador.	41
7.4	Diagrama de casos de uso correspondente á xestión de tarefas do administrador.	41
7.5	Diagrama de casos de uso correspondente á xestión de rutinas do administrador.	42
7.6	Diagrama de casos de uso correspondente á xestión de estatísticas do adminis- trador.	42
7.7	Diagrama de casos de uso correspondente ao usuario.	43
7.8	Diagrama de fluxo da aplicación.	45
7.9	Modelo JSON que representa a base de datos non relacional.	46
8.1	Modelo conceptual do patrón Modelo Vista Controlador.	49
8.2	Bosquexo correspondente ao rexistro inicial do usuario.	52
8.3	Bosquexo correspondente ao login inicial do usuario.	52
8.4	Bosquexo correspondente a mostrar tarefas restantes e coma funcionarían as notificacións da aplicación.	53
8.5	Bosquexo correspondente á acción de autenticación do administrador.	54
8.6	Bosquexo correspondente ao banco de tarefas e ao banco de rutinas.	54

8.7	Bosquexo correspondente á eliminación dunha tarefa do banco de tarefas.	55
8.8	Bosquexo correspondente a engadir tarefas.	56
8.9	Bosquexo correspondente a engadir rutina.	56
8.10	Bosquexo correspondente á visualización dos diferentes tipos de estatísticas.	57
8.11	Bosquexo correspondente a engadir novos administradores.	57
8.12	Representación en árbore da organización do código do proxecto	59
9.1	Captura explicativa da composición de widgets na listaxe das tarefas.	62
9.2	Capturas da procura de pictogramas correndo na aplicación real.	66
9.3	Capturas da listaxe de tarefas dentro da rutina correndo na aplicación real.	67
9.4	Capturas das estatísticas correndo na aplicación real.	70

Índice de Táboas

3.1	Táboa comparativa das características das aplicacións existentes.	12
6.1	Custo dos recursos humanos en euros por hora.	33
6.2	Custo dos recursos materiais/software en euros.	33
6.3	Custo total do proxecto en euros.	33
8.1	Colección correspondente aos datos do/s administrador/es.	48
8.2	Colección correspondente aos datos do/s usuario/s.	49
8.3	Subcolección correspondente aos datos da/s tarefa/s.	49
8.4	Subcolección correspondente aos datos da/s rutina/s.	50
8.5	Subcolección correspondente aos datos da/s tarefa/s dentro da/s rutina/s. . . .	50
8.6	Subcolección correspondente aos datos da/s tarefa/s dentro da/s rutina/s. . . .	51
11.1	Táboa comparativa das características das aplicacións existentes.	78

Introdución

O presente traballo consiste na implementación dunha aplicación móbil para a realización e seguimento de rutinas en nenos/as con TEA. Nas seguintes seccións desta introdución explícase o significado das siglas TEA, a motivación que me levou á realización deste traballo, os obxectivos do mesmo e a súa estrutura.

Esta introdución é necesaria para comprender a uns niveis básicos en que consiste o perfil do usuario final cara ao que está orientado este traballo e cal é a situación actual na que se atopan estes nenos.

1.1 Trastorno de Espectro Autista (TEA)

TEA son as siglas de Trastorno do Espectro Autista. O TEA é un trastorno neuro-biolóxico do desenvolvemento que se manifesta durante os tres primeiros anos de vida e que durará todo o ciclo vital.

Existen moitos tipos de TEA, e todos eles son moi complexos e variables na súa presentación e manifestación clínica [2]. A característica que mellor define ás persoas con TEA é a presenza dun deterioro distintivo na natureza e calidade do desenvolvemento social e comunicativo (influenciado polas circunstancias biolóxicas e ambientais específicas do individuo).

Normalmente estas persoas teñen unha maior dificultade á hora de aprender habilidades sociais e comunicativas pero, coa axuda dalgúns métodos de ensino concretos, que se caracterizan maioritariamente por facer explícita a información social, adquiren unha maior facilidade para asimilar esa información. Un destes métodos concretos é o emprego de pictogramas. Os pictogramas son símbolos baseados en debuxos que demostran accións concretas [3]. Neste Traballo Fin de Grao (TFG) usaremos pictogramas na interface de usuario da aplicación para o perfil dunha persoa con TEA.

1.2 Motivación

Segundo a Organización Mundial da Saúde (OMS), un de cada 160 nenos ten algún tipo de TEA e, como se menciona no apartado 1.1, calquera trastorno do espectro autista permanece durante todo o ciclo de vida do individuo, dando lugar así a situacións de aprendizaxe social complicadas especialmente durante a súa infancia e adolescencia.

Por outro lado, nos últimos anos pódese observar un auxe no uso dos *smartphones* e a idades cada vez máis temperás. De feito, no caso de España, dous de cada tres nenos de entre 10 e 15 anos teñen un *smartphone* persoal [4].

Tendo en conta estes datos, por unha parte o número de nenos que sofren algún TEA e, por outra, a idade á que os nenos comezan a ter un *smartphone* persoal, pódese concluír que este tipo de dispositivos constitúe unha ferramenta de grande utilidade para explotar no ámbito do tratamento e intervención ante un TEA.

Este traballo xorde a partires da iniciativa do propio estudante e froito dun interese persoal logo de traballar en varias ocasións con persoas con algún tipo de TEA. Ademais, esta circunstancia permitiulle ao estudante entrar en contacto con educadores que traballan dentro deste campo e observar as necesidades e falta de recursos que existen na contorna dos TEA.

1.3 Obxectivos

O principal obxectivo deste proxecto é a creación dunha aplicación nativa para os sistemas operativos móbiles Android e iOS que debe estar formada por dúas partes moi diferenciadas: unha para o usuario final da aplicación, que será unha persoa con TEA, centrándose no seu uso por nenos de corta idade; e outra para as persoas encargadas do seu coidado. As características de cada unha destas partes teñen que ser as seguintes:

- Características da parte orientada aos usuarios:
 - Cumprir como axenda persoal diaria, adaptada ás necesidades das persoas con TEA. A interface debe estar pouco cargada e orientada ás tarefas ou actividades que serán mostradas a través de pictogramas.
 - Permitir ao usuario ver un resumo de todas as tarefas do día.
 - Permitir ao usuario seleccionar a finalización correcta dunha tarefa ou a non finalización da mesma.
 - No caso de dúbidas coa realización dunha tarefa, mostrala desagregada en pequenas subtareas.

- Características da parte orientada ás persoas encargadas do coidado do usuario con TEA:
 - Acceso a esta parte da aplicación mediante algún tipo de identificación (*login*), podendo permitir o acceso a diferentes usuarios.
 - Permitir a edición de diferentes tipos de rutinas para o usuario final.
 - Permitir a introdución de novos pictogramas así como imaxes propias para a definición de novas tarefas.
 - Realizar un seguimento das tarefas completadas e non completadas do usuario final.
 - Permitir a consulta de resultados estatísticos sobre as tarefas realizadas pola persoa con TEA.

Para conseguir un *feedback* sobre a experiencia de usuario, ao longo do ciclo de desenvolvemento do proxecto, facilitarase unha versión de proba ou acceso a unha versión actualizable da aplicación a educadores e a persoas diagnosticadas con TEA de asociacións de TEA de A Coruña. Esta realimentación servirá para a mellora da aplicación, permitindo coñecer que elementos da mesma poderían modificarse ou eliminarse para o seu uso efectivo con nenos con TEA.

1.4 Estrutura

A estrutura desta memoria segue unha orde natural da evolución do proxecto. Desta forma, a memoria está composta polos seguintes capítulos:

Introdución. É o presente capítulo, correspondente á parte inicial deste TFG.

Trastorno do Espectro Autista. Capítulo explicativo sobre os trastornos do espectro autista, no que se inclúen tamén os seus tipos máis comúns e a evolución histórico deste trastorno.

Estado da Arte. Capítulo de comparación e análise das ferramentas e utilidades actualmente dispoñibles no mercado que comparten similitudes co presente TFG.

Fundamentos e Ferramentas Tecnolóxicas. Capítulo no que se explican as ferramentas tecnolóxicas utilizadas para a realización deste TFG.

Metodoloxía. Capítulo descritivo da metodoloxía e da súa aplicación para o desenvolvemento deste traballo.

Planificación e Seguimento. Capítulo onde se explica a planificación inicial de tarefas do proxecto, os recursos necesarios e os custos derivados do desenvolvemento, e onde se detalla o seguimento do mesmo.

Análise de Requisitos. Capítulo que describe a análise previa dos requisitos, das características e das funcionalidades que a aplicación debe cumprir unha vez rematada.

Deseño. Capítulo que amosa o deseño da solución analizada no capítulo anterior cos seus correspondentes diagramas.

Implementación. Capítulo utilizado para expoñer as partes máis importantes da implementación deste TFG.

Probas. Capítulo orientado ás probas realizadas para a correcta solución de erros posterior á implementación e no que se recolle o *feedback* dos usuarios que probaron a aplicación.

Conclusións. Capítulo onde se detallan as conclusións deste TFG e se propoñen liñas de traballo para o futuro.

Trastorno do Espectro Autista

NA páxina da *Autism Society of America* consta que desde os anos 90 houbo un incremento do 172 % de casos de autismo diagnosticado [5]. Isto é debido a que, antigamente, estes casos eran clasificados como “*mentally retarded*” debido ao descoñecemento das características dos TEA. En España, segundo a última enquisa do Instituto Naional de Estadística (INE) do ano 2008 con respecto ás persoas que padecen algún tipo de autismo ou outros trastornos asociados ao autismo [6], un 3.3 % da poboación en España sofre dalgún TEA, mentres que a anterior enquisa correspondente ao ano 1999 englobaba estes tipos de trastornos dentro dos retrasos mentais [7].

Este recente incremento do número de persoas diagnosticadas con algún tipo de TEA e o interese da comunidade científica e educativa provocou o aumento do seu estudo e, polo tanto, unha maior atención nos últimos anos a este tipo de trastornos. Un exemplo desta maior atención é a realización dunha mostraxe por parte da Confederación de Autismo de España con moita máis frecuencia nos últimos anos que en anos previos. Ademais, este interese emergente provocou a aparición de novas investigacións, organizacións e programas educativos orientados unicamente aos TEA. A nivel nacional, pódense atopar a Confederación de Autismo de España [8], a Asociación de Padres de Personas con Autismo [9] ou a Federación Española de Autismo [10]. A nivel internacional, destacan a *Autism Society of America* [5], a *World Autism Organization* [11] ou a asociación *Autism Speaks* [12], entre moitas outras.

Tamén se realizaron avances no ámbito educativo, da investigación e médico como, por exemplo, o artigo de 2019 co título de *Special Education Versus Inclusive Education: The Role of the TEACCH Program* [13] no contexto educativo. No ámbito de investigación pódese destacar o traballo titulado *Autism as a Model of Abnormal Emotional Development* [14]. Finalmente, no ámbito médico pódese destacar o artigo *Detailed Assessment of Incontinence, Psychological Problems and Parental Stress in Children with Autism Spectrum Disorder* [15].

2.1 Revisión histórica

Hai un século para a nosa sociedade os TEA non existían. Aínda que estaban presentes, posto que estes trastornos si existían, non eran recoñecidos como tal. Unha persoa con algún tipo de TEA era diagnosticada directamente dun problema de desenvolvemento neurolóxico. As dúas primeiras persoas que aludiron explicitamente ao autismo foron Leo Kanner (1894–1981) [16] e Hans Asperger (1906–80) [17] a principios dos anos 40. Isto ocorre no transcurso da Segunda Guerra Mundial, motivo polo que, ata a recuperación da mesma (finais dos anos 50 e principios dos 60), non se recoñecen os primeiros casos por familias e profesionais.

A primeira toma de contacto do mundo científico con algún tipo de TEA foi co primeiro estudo publicado por Leo Kanner en 1943, *Autistic Disturbances of Affective Contact* [18]. Este artigo estuda o comportamento de once nenos con algún tipo de TEA e busca identificar os trazos comúns destes comportamentos.

2.2 Visión xeral sobre o TEA

Os TEA son altamente complexos e variables nas súas manifestacións. Os síntomas e comportamentos habituais relacionados con estes trastornos dependen da madurez, grao do trastorno e do deterioro cognitivo asociado que presenta o paciente. Estas características son as que fan, en moitas ocasións, que os procesos de diagnóstico sexan difíciles. Con todo, a característica principal destes trastornos é a presenza dunha discapacidade na naturalidade e calidade do desenvolvemento social e comunicativo. Esta falta de capacidade en ambos eixos é a que distingue os TEA claramente doutras condicións de neurodesenvolvemento, coma poden ser dificultades da aprendizaxe ou trastornos no desenvolvemento da linguaxe.

As dificultades de aprendizaxe están asociadas co baixo rendemento nunha ou máis áreas, como poden ser ler, deletrear, escribir e/ou matemáticas. En xeral, as dificultades de aprendizaxe son de orixe neurolóxico permanente, o quere dicir que non se poden intervir [19].

Por outro lado, os trastornos no desenvolvemento da linguaxe están relacionados coa dificultade sintáctica que sofren certas persoas. Estes trastornos poden existir desde o momento do nacemento da persoa ou poden verse desencadeados por unha causa externa, como podería ser un dano físico [20].

Outras características que habitualmente presentan as persoas con algún tipo de TEA son a rixidez nos procesos cognitivos, intereses moi limitados ou dificultade para enfrontarse a cambios non anticipados na súa rutina diaria, entre outras.

2.2.1 Cando comeza un TEA?

Os trastornos mentais que se deben a causas xenéticas e presentes desde a infancia coñécense como trastornos do neurodesenvolvemento [2]. Os TEA categorízanse como trastornos do neurodesenvolvemento, xa que se orixinan antes do nacemento da persoa por causas xenéticas. Esta orixe dá lugar a unha maneira de funcionamento de todo o sistema nervioso central completamente distinto ao dunha persoa que non sofre de ningún TEA.

2.2.2 Cando se pode diagnosticar un TEA?

Polo xeral, as características dun TEA soen presentarse nun individuo no seu segundo ano de vida, exceptuando a Síndrome de Asperxer ou a Síndrome de Heller, nas que as primeiras manifestacións serán máis tardías [2]. O feito de que os primeiros síntomas dos TEA se produzan xa nestas idades temperás explícase porque esta etapa do crecemento constitúe o punto crítico para o inicio do comportamento social humano [2].

A día de hoxe non hai indicacións de que os TEA sexan trastornos que se poidan curar, como ocorre con calquera enfermidade común. Pola contra, os TEA son unha condición cunha base xenética e con moitas variedades, é dicir, sen unhas liñas perfectamente definidas. Por isto, o que se fai a día de hoxe é achegar ferramentas para facilitar o desenvolvemento de habilidades para tratar con situacións difíciles que as persoas que sofren dalgún tipo de TEA viven na súa rutina diaria.

2.2.3 Que trazos característicos teñen as persoas con TEA?

As principais características dunha persoa con TEA son as seguintes:

- Problemas na reciprocidade social (o dar e recibir nunha interacción social).
- A integración verbal e non verbal dentro dunha conversación.
- O desenvolvemento da amizade ou relacións de tipo social.
- Problemas para compartir sensacións de interese, empatía e curiosidade con outras persoas.
- Evasión persoal da realidade.
- Resposta non correspondente a un estímulo externo (olores, sabores, luces, etc.)
- Dificultade para comprender indicacións ou situacións pouco estruturadas.
- Realización de actividades de pouco alcance de xeito repetitivo, por exemplo dar voltas sobre si mesmos, balancearse, facer o mesmo son durante un tempo...

- Dificultade para enfrontarse a cambios imprevistos na súa rutina diaria.
- Posibilidade de manifestación de condutas agresivas.

Todas estas dificultades e a tendencia a dispersarse dunha persoa con TEA provocan que teñan maior facilidade para asimilar accións descritas a través de pictogramas, en lugar do uso de instrucións verbais, xa sexan orais ou escritas. Ademais, ás persoas con TEA non lles gustan os cambios e por iso é importante anticipar o que vai pasar mediante rutinas moi estruturadas. Cando unha persoa con TEA ten que enfrontarse a unha tarefa que non lle gusta facer, anticipar esta tarefa e avisar de que vai ter que facerse, axuda a reducir o estrés da persoa. Este é o motivo polo que se soen utilizar axendas “manuais” cos pictogramas que representan as tarefas que se van a realizar ao longo do día. Deste xeito, a persoa con TEA pode visualizar cales son as seguinte tarefas que se lle van a presentar.

Polo tanto, por todo o explicado anteriormente, queda claro que traballar con rutinas ten unha relevancia especial para as persoas con TEA.

Por outro lado, a utilización da aplicación que se vai a implementar con este TFG, no canto da maneira “manual” tradicional de xestión destas tarefas, permite a dixitalización das mesmas coas súas consecuentes vantaxes, como poden ser a automatización de tarefas, flexibilidade e facilidade de programar rutinas compostas de tarefas e a posibilidade de recoller datos de uso dun xeito automático para facilitar o seguimento das rutinas e mellorar o deseño de posibles intervencións.

2.2.4 Que se inclúe na categoría dos TEA?

Hans Asperger foi das primeiras persoas en recoñecer que existen diferentes variedades de TEA, incluíndo algunhas apareladas cunha grande intelixencia, como foi a síndrome que posteriormente herdaría o seu nome. Tamén foi unha das primeiras persoas en describir o autismo non só en nenos, senón tamén en adultos [17].

En maio de 2019, a OMS presentou a CIE-11[21]. Nesta nova clasificación internacional das enfermidades inclúense os seguintes tipos de TEA na categoría correspondente:

- *O autismo ou TEA común*

O autismo, segundo a CIE-11[21], é característico por presentar déficits persistentes na capacidade de iniciar e soste a interacción social recíproca e a comunicación social, e tamén por un rango de patróns de comportamento e intereses restrinxidos, repetitivos e inflexibles.

- *A síndrome de Asperger*

A investigación desta síndrome gañou moita popularidade nos últimos anos e é característica por manifestarse nunha idade máis tardía (sobre os oito anos, pode que mesmo máis tarde).

A síndrome de Asperxer está marcada pola manifestación dunha gran intelixencia xunto coas características comúns dos TEA mencionadas no punto anterior [22].

- *A síndrome de Heller*

A síndrome de Heller ou trastorno desintegrativo infantil tamén pertence ao espectro autista, pero é moito menos habitual que a Síndrome de Asperxer ou un TEA común.

Comparte as características dun TEA e defínese pola perda de coñecementos que os nenos adquiriron previamente. Esta síndrome é recoñecible a unhas idades un pouco máis tardías que un TEA común, en concreto un ou dous anos despois [23].

- *Outros trastornos xeneralizados do desenvolvemento, non especificados.*

Estado da Arte

A día de hoxe existen poucas aplicacións similares á implementada neste TFG. A continuación, expóñense as características das aplicacións existentes máis similares e dispoñibles no mercado.

3.1 Día a Día

Día a día [24] é un diario visual pensado para persoas con TEA ou con dificultades de comunicación. Desenvolveuse pola Fundación Orange e BJ Adaptaciones. A aplicación caracterízase por dar especial relevancia aos elementos visuais. É multiplataforma para os sistemas iOS e Android. Segundo as opinións dos usuarios, as carencias desta aplicación son que a xestión das tarefas é engorrosa e que hai unha importante falta de pictogramas.

3.2 In-TIC Agenda

In-TIC Agenda [25] é unha ferramenta destinada a persoas con TEA que emula o comportamento das axendas convencionais, e permite aos usuarios completar a súa axenda e estruturar as actividades que realizan ao longo do día. Esta aplicación unicamente está dispoñible para ordenadores, en concreto para o sistema operativo Windows. Esta aplicación si ten unha gran cantidade de pictogramas, permitindo a definición de rutinas por días pero non realiza un seguimento das mesmas nin das súas tarefas.

3.3 Diario de Autismo

Diario de Autismo [26] é unha aplicación encargada de rexistrar todos os elementos chave das vidas das persoas con TEA. Esta aplicación divide o día en catro partes diferenciadas para unha mellor visión de todo o día. Caracterízase por ter unha moi boa e acertada interface e

permite a consulta de datos estatísticos. Pola contra, esta aplicación céntrase máis na saúde física do usuario que nas rutinas ou tarefas, sendo a finalidade da mesma o seguimento de comidas, nutrientes, suplementos, medicación, etc., que o usuario precisa no seu día a día.

3.4 Choiceworks

Choiceworks [27] é unha aplicación que se centra en manter a atención do usuario cun cronómetro mentres lle amosa a tarefa que está a realizar e en como debe facela, mediante un pictograma sempre en pantalla. O punto forte desta aplicación é a liberdade que lle dá ao usuario de elixir cal é a seguinte acción que quere realizar.

Esta aplicación serviu como referencia no xeito de amosar a información ao neno, que nesta aplicación é clara e concisa. Por outra parte, a idea do cronómetro que implementa esta aplicación pode non resultar axeitada para determinados tipos de TEA, xa que pode causar tensión o feito de estar vendo continuamente un cronómetro que vai reducindo o seu tempo.

De todos os xeitos, esta aplicación só está dispoñible para iOS e leva anos sen actualizarse, deixando así sen utilizar as características novas dos *smartphones*.

3.5 Pictorario

Pictorario [28] é unha aplicación que funciona coma axenda virtual en forma de reloxo analóxico combinado con pictogramas. Permite separar cada actividade con cores e pictogramas modificando así a aparencia do reloxo analóxico. Este reloxo analóxico é a principal característica da aplicación que, combinado coas cores e os pictogramas, resulta fácil de comprender.

3.6 Táboa comparativa

Tendo en conta as principais funcionalidades que debe ter a aplicación que se vai implementar neste traballo, a táboa 3.1 amosa as características incorporadas por cada unha das aplicacións mencionadas nos apartados anteriores.

Característica Aplicación	Inclusión de Pictogramas	Definición de Rutinas	Seguimento de Tarefas	Consulta de Estatísticas	Sistema Operativo	Prezo
<i>Día a Día</i>	✓	✓	✗	✗	iOS e Android	Gratuíta
<i>In-Tic Agenda</i>	✓	✓	✗	✗	Windows	Gratuíta
<i>Diario de Autismo</i>	✗	✗	✗	✓	Android	Gratuíta
<i>ChoiceWorks</i>	✓	✓	✓	✗	iOS	9.99\$
<i>Pictorario</i>	✓	✓	✗	✗	Android	Gratuíta

Táboa 3.1: Táboa comparativa das características das aplicacións existentes.

Polo tanto, á vista desta táboa, non existe unha aplicación gratuíta no mercado que nos permita alcanzas os obxectivos indicados no Capítulo 1.

A aplicación que se vai desenvolver neste TFG debe permitir a definición de rutinas adaptadas a cada usuario, que teñen que ser descritas mediante diferentes tarefas centradas en pictogramas. Ademais, debe permitir realizar un correcto seguimento das accións programadas coa obtención de estatísticas relevantes sobre as tarefas realizadas.

Fundamentos e Ferramentas Tecnolóxicas

NESTE capítulo descríbense os fundamentos e ferramentas tecnolóxicas que se empregarán para a realización deste TFG.

4.1 Colección de pictogramas de ARASAAC

ARASAAC é unha marca do Gobierno de Aragón, inscrita na Oficina Española de Patentes e Marcas. A colección de pictogramas ARASAAC [29] é unha colección moi completa de pictogramas para a comunicación aumentativa e alternativa que está baixo unha licenza Creative Commons.

Os destinatarios dos recursos gráficos ofrecidos por ARASAAC son persoas con autismo, discapacidade intelectual, descoñecemento do idioma, persoas maiores, etc.

A decisión de incluír esta colección de pictogramas para a comunicación débese a que é o conxunto de pictogramas máis utilizado tanto polas diferentes asociacións que traballan con persoas con TEA como polas familias con algún integrante da mesma con este trastorno, polo que os nenos que utilicen a aplicación xa estarán familiarizados con eles.

O proxecto está financiado polo Departamento de Educación Cultura e Deporte do Gobierno de Aragón e coordinado pola Dirección Xeral de Innovación e Formación Profesional deste departamento.

Un exemplo de diferentes pictogramas desta colección está representado na figura 4.1.



Figura 4.1: Exemplo de pictogramas da colección de ARASAAC.

4.2 IntelliJ IDEA

IntelliJ [30] é un Entorno de Desenvolvemento Integrado (IDE) promovido por JetBrains e dispón de licenza para estudantes para a súa versión *Ultimate Edition*. Este IDE proporciona unha moi boa integración con Dart, vía *plugins*, así como unha maneira moi sinxela de executar e depurar as aplicacións realizadas con esta linguaxe de programación.

4.3 Android Studio

O programa *Android Studio* [31] está baseado no IDE mencionado anteriormente (IntelliJ) pero con maiores funcionalidades para a integración co sistema operativo móbil Android.

Entre as devanditas funcionalidades atópanse todos os controladores encargados de realizar a conexión entre o computador e o dispositivo sobre o que correrá a aplicación, neste caso, un emulador Android tamén integrado en Android Studio e baseado nun terminal xenérico que será o utilizado para probar a aplicación.

Deste xeito, esta ferramenta é utilizada neste TFG para dispoñer de todos os controladores encargados da conexión con terminais Android externos e, así, poder probar a aplicación nalgún dispositivo físico. Ademais, é utilizado para poder correr un emulador Android oficial que se utilizará no desenvolvemento da aplicación.

4.4 Xcode

O programa *Xcode* [32] é un IDE para macOS que contén un conxunto de ferramentas creadas por Apple destinadas ao desenvolvemento de software para o seu propio hardware. A importancia de Xcode neste proxecto é o uso do emulador de dispositivos móbiles propio de Apple, así como a inclusión de todos os controladores necesarios para que a aplicación corra sobre dispositivos iOS.

4.5 Dart

Dart [33] é unha linguaxe de programación desenvolvida por Google para múltiples plataformas. Trátase dunha linguaxe multiparadigma, aínda que se usa principalmente seguindo o paradigma da orientación a obxectos. Permite polo tanto definir clases e contén un recolector de lixo propio. Ten unha sintaxe similar a C.

Dart é unha linguaxe de programación optimizada para crear interfaces de usuario e é capaz de realizar recargas asíncronas dunha única parte da interface, sen necesidade de actualizar todas os seus compoñentes. Tamén permite facer cambios iterativamente no código visualizándoos instantaneamente no dispositivo sobre o que se estea probando o software.

Dart usa illamento para xestionar a concorrencia. Deste xeito, os procesos concorrentes non comparten memoria e no seu lugar usan intercambio de mensaxes (similar a como xestiona a concorrencia Erlang).

O uso de Dart neste proxecto ven motivado porque é a linguaxe de programación utilizada para desenvolver aplicacións móbiles nativas con Flutter.

4.6 Flutter

Flutter [34] é un software de desenvolvemento de interfaces de usuario de código aberto creado por Google. É usado para crear aplicacións nativas para Android, iOS, Windows, Mac, Linux, Google Fuchsia e web.

A primeira versión de Flutter foi lanzada en 2015 e unicamente era compatible co sistema operativo Android nos seus inicios. Sen embargo, foise estendendo a súa compatibilidade co resto de plataformas a medida que se lanzaron novas versións.

A gran vantaxe de usar Flutter é a posibilidade de crear aplicacións nativas para diferentes plataformas a partires dun único código fonte con moi pequenas variacións. Isto supón unha gran vantaxe neste proxecto tendo en conta que se quere desenvolver unha aplicación multiplataforma.

Flutter destaca por usar composición, é dicir, unha interface de usuario desenvolvida con Flutter estará composta de *widgets*, que á súa vez estarán compostos de máis *widgets* dentro dos mesmos e así sucesivamente. Todos estes *widgets* inicianse co método `build()`. Isto permite definir vistas máis compactas e a reutilización de compoñentes en diferentes pantallas.

4.7 Firebase

Firebase [35] é unha plataforma de Google, ubicada na nube, que utiliza un conxunto de ferramentas para a creación e sincronización de proxectos de distinto tipo. A configuración e

integración de Firebase neste proxecto será a encargada da xestión do *backend* da aplicación, facendo posible o crecemento do número de usuarios, a escritura e lectura de arquivos na nube e a xestión dunha base de datos para o correcto funcionamento da aplicación. Ademais, neste proxecto utilízanse unha serie de tecnoloxías moi útiles que proporciona Firebase e que son descritas a continuación.

4.7.1 Cloud Firestore

Cloud Firestore [36] é unha base de datos NoSQL flexible, escalable e aloxada na nube co obxectivo de almacenar e sincronizar datos para a programación no lado do cliente e do servidor. A funcionalidade chave de Cloud Firestore é que mantén os datos sincronizados entre apps cliente a través de axentes de escoita en tempo real.

4.7.2 Google Cloud Storage

Google Cloud Storage [37] é unha opción de almacenamento na nube, escalable e duradeiro. Utilízase para ler e escribir calquera tipo de arquivos en depósitos de Cloud Storage. Na aplicación utilízase para almacenar e acceder na nube ás imaxes propias (xa sexan da cámara ou do almacenamento propio do dispositivo onde se corre a aplicación), para o seu posterior uso no mesmo dispositivo ou noutros.

4.7.3 Firebase Authentication

Firebase Authentication [38] proporciona servizos de *backend* xa elaborados para facilitar a autenticación dos usuarios da aplicación. Admite a autenticación mediante diferentes factores aínda que, no caso desta aplicación, utilízase o par correo electrónico máis contrasinal.

Grazas á combinación desta funcionalidade de Firebase coa de Cloud Storage 4.7.1 é posible ter simultaneamente varios usuarios administradores compartindo e traballando sobre os mesmos datos do neno.

4.8 Android

Android [39] é un sistema operativo móbil de código aberto baseado nunha modificación do kernel Linux. Actualmente, xunto con iOS, ten o dominio no mercado de sistemas operativos para dispositivos móbiles.

Android foi desenvolvido entre varias compañías e institucións, entre as que destacan Google e The Open Handset Alliance. Aínda que é un sistema operativo de código aberto, polo xeral as versións ás que estamos acostumados teñen incluído software privativo, como pode ser calquera dos servizos de Google (tenda de aplicacións, mapas, buscador, etc.)

Debido á súa versatilidade e ao seu carácter de código aberto, podemos atopar o sistema Android en moitos dispositivos diferentes, desde móbiles ata reloxos, pasando por coches, navegadores GPS, radios, etc.

4.9 iOS

iOS [40] é un sistema operativo móbil creado e desenvolvido por Apple exclusivamente para o seu propio hardware. Actualmente é o sistema operativo que corre baixo os iPhone, iPod Touch e iPad (dispositivos móbiles que ofrece a compañía Apple).

Xunto con Android, coma se mencionou anteriormente, ten o dominio do mercado de sistemas operativos para dispositivos móbiles, destacando por estar en primeiro lugar no uso de sistemas operativos móbiles en tabletas cos iPads de Apple.

4.10 LATEX

LaTeX [41], ás veces escrito como LATEX, é un sistema de composición tipográfica. LaTeX inclúe características especificamente deseñadas para a produción de documentación técnica e científica. Por esa razón, LaTeX é o estándar para a comunicación e publicación de documentos científicos. Ademais, está dispoñible como software libre.

Esta ferramenta permite escribir documentos, cuxo deseño se realiza automaticamente axustándose da mellor maneira posible aos estándares tipográficos. Unha característica distintiva de LaTeX é o seu modo matemático, que permite compoñer e representar fórmulas complexas.

4.11 Git

Git é un software de control de versións [42]. A súa funcionalidade é levar un control dos cambios realizados durante o desenvolvemento da aplicación software.

Git utilizouse neste proxecto para levar un control das funcionalidades implementadas, podendo consultar implementacións realizadas previamente no caso de ser necesario e mesmo regresar a algún estado da implementación pasado necesario nalgún momento puntual.

4.12 Microsoft Project 2016

Microsoft Project 2016 é a versión do programa de Microsoft Project [43] que se utilizou neste proxecto. Microsoft Project é un produto software de xestión de proxectos desenvolvido por Microsoft e pertencente ao conxunto Microsoft Office. Está deseñado para axudar a un xefe

de proxecto a desenvolver un cronograma, asignar recursos a tarefas, rastrexar o seu progreso, administrar o orzamento e analizar as cargas de traballo.

4.13 Visual Paradigm Online Diagrams

Visual Paradigm Online Diagrams [44] é un conxunto de ferramentas de modelado de diagramas que está dispoñible online dunha maneira gratuíta para un uso non comercial. A súa primeira versión estable foi lanzada ao público en 2018.

Ademais de ser unha ferramenta para a realización de diagramas tamén é capaz de auto-xerar os diagramas con enxeñería inversa a partires de código escrito en diferentes linguaxes de programación.

Esta ferramenta utilizouse para xerar todos os diagramas e esquemas da aplicación que se achegan nesta memoria.

4.14 Balsamiq Wireframes

Balsamiq Wireframes [45] é unha ferramenta que pode ser web ou aplicación de escritorio e que permite a creación de *wireframes* e *mockups* (bosquexos) a partires de *widgets* predefinidos. O deseño con esta aplicación realízase cun sistema de arrastrar e soltar, resultando así unha ferramenta moi sinxela e cómoda para realizar bosquexos.

Os bosquexos de deseño previos á implementación da aplicación que se achegan na memoria realizáronse con esta ferramenta.

Metodoloxía

A elección dunha metodoloxía axeitada para levar a cabo o desenvolvemento dun produto software resulta moi importante xa que esta permite definir regras explícitas e procedementos para afrontar os desafíos que supón un proxecto deste tipo. Neste caso, a metodoloxía elixida para este TFG foi unha metodoloxía iterativa incremental baseada en SCRUM [46], onde as correspondentes iteracións (denominadas *sprints*) terán unha duración fixa de 15 días. Aínda que SCRUM está ideado para que os proxectos sexan levados a cabo por equipos de traballo, con diferentes roles dentro dos mesmos, neste caso estes roles serán asumidos por unha única persoa.

Nas seguintes seccións imos describir os principios nos que se asenta a metodoloxía SCRUM e os diferentes eventos e elementos que a definen.

5.1 Definición

SCRUM é unha metodoloxía de software áxil utilizada comunmente para o desenvolvemento de software. Para a correcta utilización da metodoloxía SCRUM é preciso definir unha serie de iteracións de tal forma que, desde o comezo, se entrega un produto funcional que, progresivamente, iteración a iteración, se vai convertendo no produto final.

SCRUM está ideado para facilitar o traballo dun equipo que interactúa cun cliente desenvolvendo un produto para uns usuarios finais. Ademais, ao tratarse dunha metodoloxía de desenvolvemento áxil, ofrece certa flexibilidade á hora de introducir cambios nos requisitos do produto debido ao *feedback* proporcionado polo cliente cando revisa o produto funcional que se entrega en cada iteración. Isto evita a necesidade de partir dunha definición de requisitos fixa e moi elaborada e, ademais, permite adaptarse a posibles cambios que poidan xurdir durante o ciclo de vida do proxecto.

5.2 Roles

Os tres roles principais que se definen dentro do marco de traballo establecido por SCRUM son os de *Product Owner*, *SCRUM Master* e *Development Team*.

5.2.1 Product Owner

É a persoa ou grupo de persoas que está do lado do cliente, coñece todas as súas necesidades e encárgase de transmitir estas necesidades ao *SCRUM Master*.

5.2.2 SCRUM Master

É a persoa responsable de asegurar que o marco de traballo definido por SCRUM é entendido e realizado dunha maneira axeitada polo equipo de desenvolvemento, axustándose ás datas e cumprindo os requisitos.

5.2.3 Development Team

É o equipo de desenvolvemento, é dicir, o conxunto de persoas capacitadas para implementar os requisitos que o cliente necesita.

5.3 Sprints

O método SCRUM divide o traballo que se vai realizar nunha serie de *sprints*. Un *sprint* correspóndese con cada iteración do produto que vai ser entregada. Deste xeito, para cada *sprint* é preciso establecer na reunión de planificación do mesmo o conxunto de casos de uso ou historias de usuario que se van abordar, realizar a implementación das historias seleccionadas e entregar un produto funcional que incorpore as correspondentes funcionalidades. Ao final do *sprint*, é preciso realizar unha revisión na que se recibe o *feedback* do *Product Owner* e se pode facer unha retrospectiva do traballo realizado. Os *sprints* caracterízanse porque cada novo *sprint* é a continuación do anterior e por manter unha duración consistente no tempo, tendo así fixadas as datas de inicio e de fin de cada *sprint*. No caso deste TFG, estableceuse unha duración de 15 días para cada *sprint* planificado.

Dentro dos sprints existe o *Daily Scrum* ou *Daily Meeting*, utilizado para a sincronización de todos os membros do *Development Team*. Dentro de cada *sprint* do método SCRUM, defínese a mesma estrutura e que consta de: análise, deseño, implementación, probas e solución de erros. Deste xeito, cada *sprint* estará formado por catro fases diferenciadas: análise, deseño, implementación, e probas e solución de erros.

5.3.1 Reunión de planificación e revisión

Nesta primeira fase do *sprint* realízase unha reunión entre o *Product Owner* e o *SCRUM Master*.

Grazas a esta reunión o *SCRUM Master* pode planificar o *sprint* xunto co *Development Team* e revisar a realimentación recibida do *Product Owner* sobre o *sprint* anterior (no caso de non ser o primeiro *sprint*).

5.3.2 Análise

Nesta fase do *sprint* analízase a necesidade existente, definindo así os requirimentos e funcionalidades que debe cumprir o software para a fin desa iteración.

5.3.3 Deseño

Nesta fase do *sprint* deseñase a solución definida na fase de análise. Para deseñar esta solución utilizaranse un diagrama de fluxo, o deseño do modelo da base de datos e o deseño de bosquejos da posible interface da aplicación.

- Diagrama de fluxo: diagrama que amosa os fluxos de traballo paso a paso.
- Deseño do modelo da base de datos: deseño das diferentes coleccións a crear cos seus datos e as súas relacións.
- Deseño de bosquejos: deseño de bosquejos conceptuais sobre a interface final coa que os usuarios da aplicación interactuarán.

5.3.4 Implementación

É a fase dedicada a levar á práctica o deseñado na fase anterior. No caso de non coñecer perfectamente as tecnoloxías utilizadas, dentro desta fase poden xurdir problemas, o que daría lugar á obriga de retroceder e facer certos cambios sobre os deseños realizados na fase anterior.

5.3.5 Probas e solución de erros

É a derradeira fase do *sprint* e está dedicada a probar o correcto funcionamento do traballo realizado na fase de implementación, abarcando todos os casos de uso posibles. No caso de atopar erros, sería preciso deseñar e implementar unha solución aos mesmos que, posteriormente, habería que probar de novo ata verificar que non haxa erros.

5.4 Artefactos

Dentro do método SCRUM, os artefactos son os elementos físicos que se producen como resultado da aplicación de SCRUM. Estes artefactos son o *Product Backlog*, o *Sprint Backlog* e os incrementos, que se explicarán a continuación. Unha representación gráfica de todos estos artefactos xuntos pode verse na figura 5.1

5.4.1 Product Backlog

O *Product Backlog* é a recompilación de calquera traballo que sexa necesario aplicar no produto, sendo así a maior fonte de información do mesmo.

Esta información obtense do *Product Owner*, reflectindo deste xeito o estado real do traballo pendente de implementar do produto e o xa realizado ata o momento de obtención do *Product Backlog*.

O *Product Backlog* debe ser xestionado polo *Product Owner* sendo a súa función a de priorizar dentro de cada etapa as accións que crea convenientes.

5.4.2 Sprint Backlog

O *Sprint Backlog* é unha listaxe de elementos nos que se traballa dentro de cada *sprint*. Por norma xeral, estes elementos compóñense de tarefas técnicas máis pequenas que son as que dan como resultado un incremento do software determinado.

Este artefacto está xestionado polo *Development Team*, permitindo así manter a transferencia dentro do desenvolvemento actualizándoo pouco a pouco durante todo o *sprint*.

Así, o *Sprint Backlog* permite organizar e saber que elementos aínda non empezaron a desenvolverse, cales si, cales xa están rematados ou cales están en proceso de ser incluídos no incremento do *sprint*.

5.4.3 Incremento

O incremento é o produto obtido ao final de cada *sprint*. Polo tanto, é a suma de todos os resultados dos diferentes traballos feitos dentro do mesmo, obtendo un resultado en forma de software, que será o produto final posto a disposición do usuario.

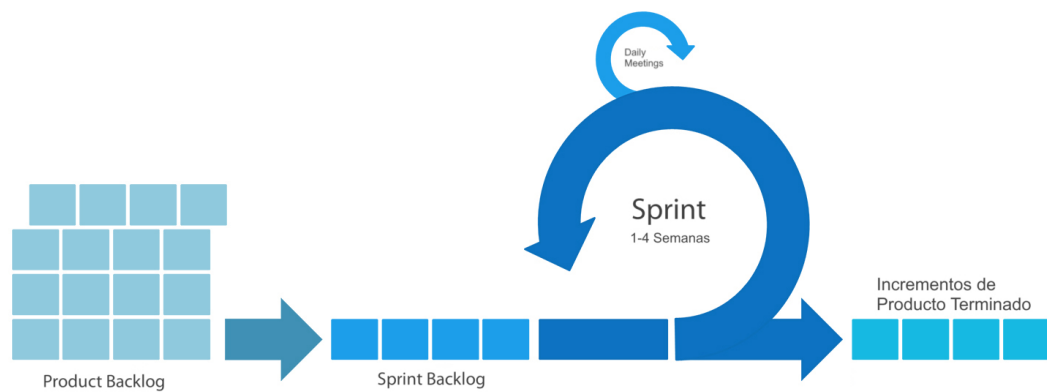


Figura 5.1: Representación gráfica sinxela de todos os compoñentes da metodoloxía SCRUM. Imaxe creada a partir da imaxe [1] baixo unha licencia Creative Commons 4.0 que permite a súa edición e uso.

Planificación, seguimento e custos

NESTE capítulo detallarase a planificación inicial establecida para a realización deste TFG, así como o seu correspondente seguimento. Ademais, levarase a cabo unha estimación dos custos asociados ao desenvolvemento do proxecto.

6.1 Recursos humanos

Para este TFG, establecéronse os seguintes tipos de perfís de recursos humanos necesarios:

- *Product Owner*.
- *SCRUM Master*.
- Analista.
- Diseñador.
- Programador.

O analista, diseñador e programador formarán o *Development Team* explicado en 5.2.3. Se ao *Development Team* lle engadimos a figura de *Scrum Master*, teríamos todos os roles que desempeña o alumno dentro deste TFG.

Por outra banda, o rol de *Product Owner* (ver 5.2.1) desempeñarano os dous titores deste TFG, completando así todos os roles que define SCRUM.

6.1.1 Product Owner

A función que desempeña o *Product Owner* dentro do proxecto é estar presente nas diferentes reunións xunto co *SCRUM Master* para realizar as revisións finais de cada *sprint* e para decidir que necesidades se van considerar durante a planificación do seguinte *sprint*.

6.1.2 SCRUM Master

O rol que desempeña o SCRUM *Master* está detallado na sección correspondente á súa definición (ver 5.2.2). Como se comentou anteriormente, este rol vai ser desempeñado polo alumno.

6.1.3 Analista

A función que desempeña o analista dentro do *Development Team* do proxecto é a de analizar as necesidades que se estableceron ao principio de cada *sprint*, ideando posibles solucións ás mesmas.

6.1.4 Deseñador

A función que desempeña o deseñador dentro do *Development Team* do proxecto é deseñar as solucións ideadas na fase de análise (ver 5.3.2). Esta persoa é a encargada de realizar os diferentes diagramas mencionados na definición da fase de deseño (ver 5.3.3), é dicir, de clases, de secuencia e de fluxo. Xunto co programador encárgase tamén da fase de probas e solución de erros (ver 5.3.5), realizando os posibles deseños das solucións.

6.1.5 Programador

A función que desempeña o programador dentro do *Development Team* do proxecto é a de implementar os deseños obtidos na fase de deseño debido ao seu coñecemento das tecnoloxías que se usarán durante o proxecto. Xunto co deseñador encárgase tamén da fase de probas e solución de erros (ver 5.3.5), implementando as solucións definidas polo deseñador.

6.2 Recursos materiais / software

A selección dos recursos materiais e de software baséase principalmente na actual dispoñibilidade dos mesmos por parte do alumno e os proporcionados pola condición de ser estudante da Universidade da Coruña.

6.2.1 Recursos materiais

Os recursos materiais utilizados para este proxecto dos que xa dispoñía o alumno son un ordenador persoal Macbook Pro cun procesador Intel Core i5 de 4 núcleos a 1,4 GHz utilizado para todo o desenvolvemento do proxecto e un iPhone XS para as probas nun dispositivo físico da aplicación final.

6.2.2 Recursos software

Os recursos software utilizados neste proxecto son os mencionados no capítulo 4 no que se especifican os fundamentos e ferramentas tecnolóxicas. Dentro destes recursos software a maneira de poder utilizar o software que non dispón de licenzas gratuítas foi:

1. Unha licenza gratuíta, por ser estudante, de IntelliJ IDEA Ultimate Edition que proporciona a propia plataforma a todos os estudantes.
2. Unha licenza proporcionada pola Universidade da Coruña de Microsoft Project 2016.
3. Unha licenza provisional dun ano concedida por Balsamiq para a utilización de Balsamiq Wireframes neste proxecto.

6.3 Planificación inicial

Para completar o desenvolvemento da aplicación planificáronse un total de cinco *sprints*, cada un dos cales ten unha duración de quince días laborables. Ademais, contemplouse unha fase final que inclúe a elaboración da documentación, que se corresponde coa finalización da presente memoria e que ten unha duración de dez días laborables, e unha revisión final do proxecto cunha duración dun día laborable. Cada *sprint* pode verse como unha tarefa que está composta, a súa vez, das súas correspondentes subtarefas (reunión, planificación e revisión, análise, deseño, implementación e probas e solución de erros).

Os obxectivos das tarefas a desenvolver para cada *sprint*, para a documentación e para a reunión final son os seguintes:

Sprint 1

- Obter e perfeccionar os coñecementos tecnolóxicos das ferramentas a utilizar durante o proxecto.
- Obter unha versión básica e non funcional da aplicación que amose como vai ser o seu deseño.
- Obter unha primeira versión da parte da aplicación correspondente ao usuario.

Sprint 2

- Crear a base de datos que vai utilizar a aplicación.
- Integrar a base de datos coa aplicación.
- Obter unha versión da aplicación con menús de navegación.
- Obter unha versión da aplicación coa parte do usuario finalizada.

Sprint 3

- Obter unha primeira versión da parte da aplicación correspondente ao administrador.

Sprint 4

- Finalización da parte da aplicación correspondente ao administrador.
- Integración da aplicación co sistema operativo.
- Conseguir unha primeira versión funcional da aplicación para a proba dos usuarios finais.

Sprint 5

- Engadir á aplicación novas funcionalidades a partires da resposta recibida polos usuarios que proben a aplicación.
- Corrixir e modificar as funcionalidades que non sexan ben acollidas polos usuarios que proben a aplicación.

Documentación

- Realización de toda a documentación do proxecto, neste caso, a presente memoria.

Reunión final

- Realización dunha reunión final para comprobar a correcta realización do proxecto.

Toda as tarefas e as subtarefas teñen entre si unha relación de Fin-Comezo. Estas relacións pódense ver representadas no diagrama de Gantt da figura 6.1.

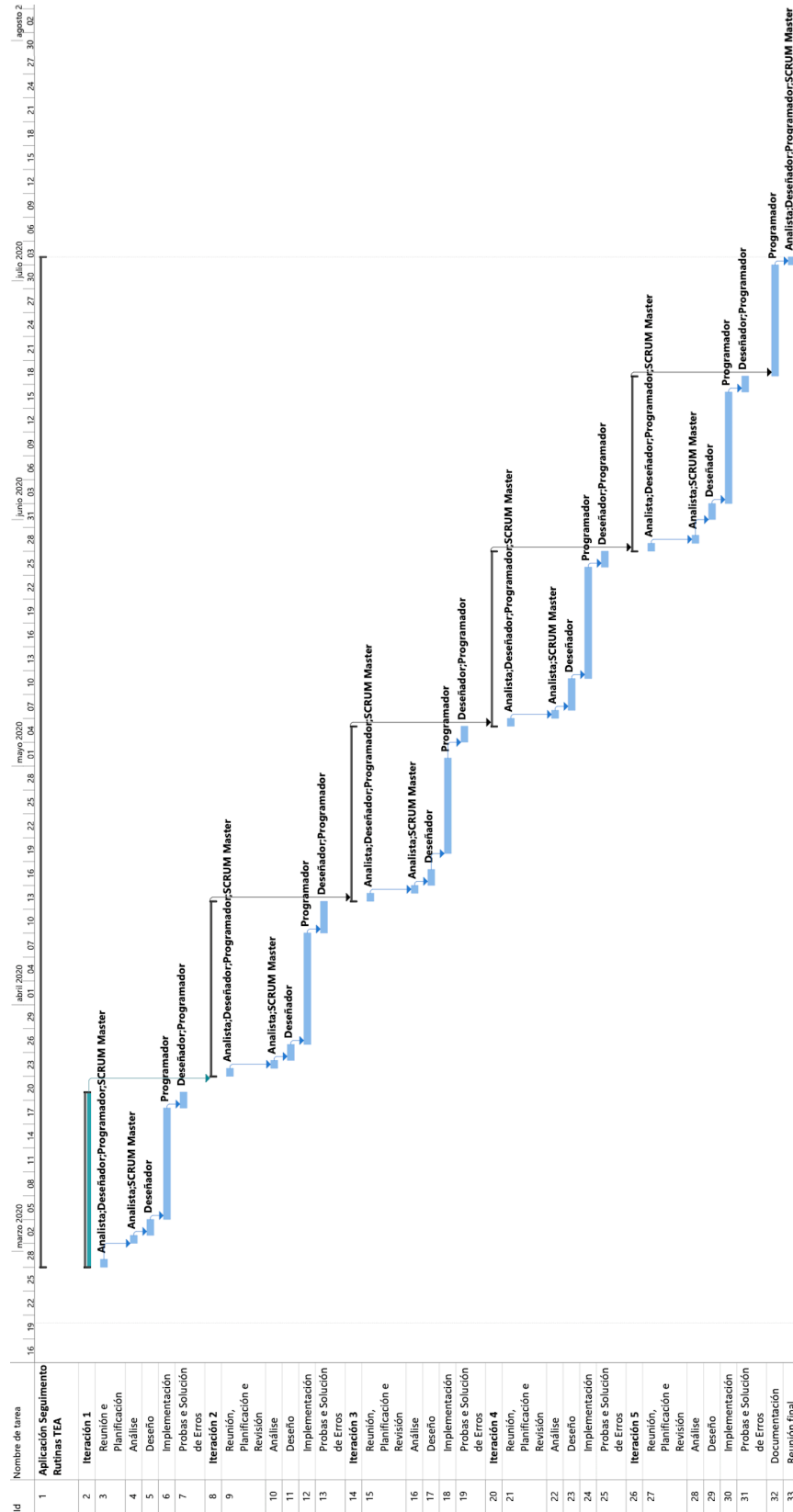


Figura 6.1: Diagrama de Gantt correspondiente á planificación.

Na figura 6.2 amósase gráficamente a planificación inicial establecida para este proxecto, onde se inclúen as diferentes tarefas que conforman cada *sprint*, a asignación de recursos a estas tarefas, a súa duración, as relacións entre as mesmas e o custo asociado a cada unha delas.

Id	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombres de los recursos	Costo
1	Aplicación Seguimiento Rutinas TEA	91 días	vie 28/02/20	vie 03/07/20			€16.640,00
2	Iteración 1	16 días	vie 28/02/20	vie 20/03/20			€2.960,00
3	Reunión e Planificación	1 día	vie 28/02/20	vie 28/02/20		Analista;Deseñador;Programador;SCRUM Master	€640,00
4	Análise	1 día	lun 02/03/20	lun 02/03/20	3	Analista;SCRUM Master	€400,00
5	Deseño	2 días	mar 03/03/20	mié 04/03/20	4	Deseñador	€240,00
6	Implementación	10 días	jue 05/03/20	mié 18/03/20	5	Programador	€1.200,00
7	Probos e Solución de Erros	2 días	jue 19/03/20	vie 20/03/20	6	Deseñador;Programador	€480,00
8	Iteración 2	16 días	lun 23/03/20	lun 13/04/20	2		€2.960,00
9	Reunión, Planificación e Revisión	1 día	lun 23/03/20	lun 23/03/20		Analista;Deseñador;Programador;SCRUM Master	€640,00
10	Análise	1 día	mar 24/03/20	mar 24/03/20	9	Analista;SCRUM Master	€400,00
11	Deseño	2 días	mié 25/03/20	jue 26/03/20	10	Deseñador	€240,00
12	Implementación	10 días	vie 27/03/20	jue 09/04/20	11	Programador	€1.200,00
13	Probos e Solución de Erros	2 días	vie 10/04/20	lun 13/04/20	12	Deseñador;Programador	€480,00
14	Iteración 3	16 días	mar 14/04/20	mar 05/05/20	8		€2.960,00
15	Reunión, Planificación e Revisión	1 día	mar 14/04/20	mar 14/04/20		Analista;Deseñador;Programador;SCRUM Master	€640,00
16	Análise	1 día	mié 15/04/20	mié 15/04/20	15	Analista;SCRUM Master	€400,00
17	Deseño	2 días	jue 16/04/20	vie 17/04/20	16	Deseñador	€240,00
18	Implementación	10 días	lun 20/04/20	vie 01/05/20	17	Programador	€1.200,00
19	Probos e Solución de Erros	2 días	lun 04/05/20	mar 05/05/20	18	Deseñador;Programador	€480,00
20	Iteración 4	16 días	mié 06/05/20	mié 27/05/20	14		€2.960,00
21	Reunión, Planificación e Revisión	1 día	mié 06/05/20	mié 06/05/20		Analista;Deseñador;Programador;SCRUM Master	€640,00
22	Análise	1 día	jue 07/05/20	jue 07/05/20	21	Analista;SCRUM Master	€400,00
23	Deseño	2 días	vie 08/05/20	lun 11/05/20	22	Deseñador	€240,00
24	Implementación	10 días	mar 12/05/20	lun 25/05/20	23	Programador	€1.200,00
25	Probos e Solución de Erros	2 días	mar 26/05/20	mié 27/05/20	24	Deseñador;Programador	€480,00
26	Iteración 5	16 días	jue 28/05/20	jue 18/06/20	20		€2.960,00
27	Reunión, Planificación e Revisión	1 día	jue 28/05/20	jue 28/05/20		Analista;Deseñador;Programador;SCRUM Master	€640,00
28	Análise	1 día	vie 29/05/20	vie 29/05/20	27	Analista;SCRUM Master	€400,00
29	Deseño	2 días	lun 01/06/20	mar 02/06/20	28	Deseñador	€240,00
30	Implementación	10 días	mié 03/06/20	mar 16/06/20	29	Programador	€1.200,00
31	Probos e Solución de Erros	2 días	mié 17/06/20	jue 18/06/20	30	Deseñador;Programador	€480,00
32	Documentación	10 días	vie 19/06/20	jue 02/07/20	26	Programador	€1.200,00
33	Reunión final	1 día	vie 03/07/20	vie 03/07/20	32	Analista;Deseñador;Programador;SCRUM Master	€640,00

Figura 6.2: Tarefas, duración, custos e recursos asignados detallados.

6.4 Custos

Para realizar unha estimación realista dos custos totais que conleva o desenvolvemento do proxecto é preciso ter en conta os gastos asociados ao soldo dos recursos humanos e os custos derivados dos recursos materiais e software que se empregaron ao longo do proxecto.

Os custos establecidos para os recursos humanos que se especifican no apartado 6.1 e incluídos na figura 6.2, son unha aproximación do salario medio por horas en España para estes traballos. Este dato pódese consultar para o SCRUM *Master* en [47], para o analista en [48], para o deseñador en [49] e para o programador en [50]. Dividindo este salario medio anual entre as 1 760 horas que se traballa anualmente (220 días laborables por ano cunha xornada de 8 horas por día), obtense o custo por hora de cada recurso humano. Os datos resultantes indícanse na táboa 6.1.

Recursos humanos \ Salario	Salario medio (€) / ano	Euros / Hora
SCRUM Master	37 008	21
Analista	34 308	19
Deseñador	18 204	10
Programador	26 821	15

Táboa 6.1: Custo dos recursos humanos en euros por hora.

Por outra parte, os custos establecidos para os recursos materiais/software detallados no apartado 6.2 son os seguintes. Sendo a duración do proxecto de catro meses e cun custo total aproximado de 1 000 euros para o ordenador portátil e de 500 euros para o teléfono móbil, obtense que o custo destes recursos materiais é de 250 euros ao mes para o ordenador portátil e de 125 euros ao mes para o teléfono móbil. Estes custos están representados na táboa 6.2.

Recursos materiais \ Prezo	Euros
Ordenador portátil	1 000
Teléfono Móbil	500
Licenza IntelliJ IDEA Ultimate	0
Licenza Microsoft Project 2016	0
Licenza Balsamiq Mockups	0

Táboa 6.2: Custo dos recursos materiais/software en euros.

Deste xeito, o custo total estimado do proxecto sería de 18 140 euros, tal e como se pode ver na táboa 6.3.

Recursos \ Custo	Euros
Humanos	16 640
Materiais	1 500
TOTAL	18 140

Táboa 6.3: Custo total do proxecto en euros.

6.5 Seguimento

Ao longo do ciclo de vida do proxecto, realizouse un seguimento do mesmo para detectar, xestionar e mitigar o impacto de posibles desviacións sobre a planificación inicial.

A maior desviación que existiu neste proxecto foi un atraso no primeiro *sprint* debido á situación persoal do alumno durante a crise sanitaria do Covid-19, atrasando así o resto de *sprints* debido ás relacións fin-comezo que existen entre elas. As subfases de análise e deseño do primeiro *sprint* realizáronse correctamente en prazo. Pero á hora de comezar coa implementación, debido á crise mencionada anteriormente e a que o alumno estaba a realizar unha mobilidade no estranxeiro, decidiuse paralizar o desenvolvemento deste TFG ata finalizar a mobilidade e poder volver a España logo das restricións motivadas pola pandemia.

Unha vez finalizada esta pausa e xa en España, a estimación da data de finalización do proxecto moveuse de finais de xuño a finais de agosto. Por outra parte, debido a que o alumno desempeña os roles dos que depende a planificación e que neste tempo o proxecto estivo paralizado, os custos estimados do proxecto mantivéronse.

Unha vez se reiniciou o proxecto, este non se viu afectado por outras desviacións significativas, rematando así fora do prazo inicialmente previsto pero mantendo os custos do mesmo.

Análise de Requisitos

NESTE capítulo vanse explicar os diferentes actores que utilizará a aplicación, os distintos requisitos que debe cumprir e os correspondentes casos de uso que se poden atopar na mesma.

7.1 Actores

En primeiro lugar, os actores que poden interactuar coa aplicación son tres: o usuario non autenticado, o usuario autenticado que non é administrador e, finalmente, o usuario autenticado que si que é administrador.

7.1.1 Usuario non autenticado

Este actor corresponde ao usuario que aínda non está identificado na aplicación.

7.1.2 Usuario autenticado

Este actor corresponde co usuario da aplicación xa autenticado pero que non ten permisos de administrador. Este actor é o neno/a con algún tipo de TEA ou un administrador que aínda non se autenticou coma tal.

7.1.3 Administrador

Este actor corresponde co usuario da aplicación que ten permisos de administrador. Este actor é un educador ou coidador que vai definir as rutinas e realizar un seguimento sobre o usuario autenticado sen permisos de administrador.

Cada administrador vai ter un único usuario asociado (o neno), pero un mesmo neno pode estar titorizado por varios usuarios administradores que poden definir e seguir as súas rutinas dunha maneira síncrona entre eles.

7.2 Requisitos

Os requisitos que debe cumprir esta aplicación poden clasificarse en requisitos funcionais e non funcionais.

Nesta aplicación, pódense categorizar os requisitos funcionais en requisitos funcionais do administrador e requisitos funcionais do usuario.

7.2.1 Requisitos funcionais (administrador)

Identificación do administrador

Permite ao administrador identificarse para poder ter acceso así a todas as tarefas administrativas da mesma. O administrador terá un contrasinal de acceso á aplicación e un contrasinal de edición para acceder ao menú de administración. Este último contrasinal pode cambiarse co seguinte requisito.

Cambiar o contrasinal de edición do administrador

Logo de loguearse o administrador dentro da súa sección de administración da aplicación, este pode cambiar o seu contrasinal de edición.

Engadir administradores

Un usuario con rol administrador poderá engadir un novo usuario administrador proporcionando os datos do novo administrador. Desta maneira os dous administradores compartirán e terán acceso ao mesmos datos do usuario non administrador.

Banco de tarefas

Todos os administradores poderán **consultar** as tarefas existentes na base de datos correspondentes ao neno do que están a facer un seguimento.

Permítese **filtrar** as tarefas polo nome para obter unha procura máis concreta.

Todos os administradores poden **engadir** novas tarefas ao banco de tarefas mediante a súa creación. Para iso, as tarefas deben ter un nome, unha imaxe (que poderá ser unha fotografía ou un pictograma da base de datos de pictogramas), unha descrición (que será unha secuencia de pictogramas representando a tarefa en pasos máis sinxelos), unha duración e un tipo (se o administrador vai querer avaliar a tarefa realizada ou non).

Tamén se pode **eliminar** calquera tarefa do banco de tarefas.

Un administrador terá acceso a todas as tarefas correspondentes ao usuario ao que está titorizando, é dicir, tamén terá acceso a tarefas que crearon outros administradores para o neno.

Banco de rutinas

Todos os administradores poderán **consultar** as rutinas existentes que foron creadas con anterioridade, podendo filtralas polo seu nome.

Todos os administradores poden **crear** ou **eliminar** rutinas. Cada rutina corresponde a un día, ten un nome único e está composta de tarefas asignadas a certos momentos do día, non podendo solaparse tarefas no mesmo momento.

Unha vez creadas as rutinas, estas poden **asignarse** a un día concreto facendo uso dun calendario integrado na aplicación.

Unha vez asignada esta rutina coas súas correspondentes tarefas diarias, o usuario que non é administrador poderá ver e interactuar con esas tarefas.

Estatísticas das tarefas

Permite ao administrador **visualizar as estatísticas das tarefas** nunha gráfica. Estas estatísticas amosan se as tarefas se realizaron ou non, a porcentaxe de tempo utilizado con respecto ao tempo previsto inicialmente para a realización da mesma, o éxito na realización da tarefa ou, pola contra, a non superación da tarefa.

As gráficas serán interactivas, permitindo a visualización dunha única estatística ou de varias na mesma pantalla. Tamén permitirán comparar tarefas na mesma gráfica para obter unha representación máis visual da comparación entre tarefas realizadas.

7.2.2 Requisitos funcionais (usuario)

Mostrar información das tarefas

A aplicación **mostra ao usuario a tarefa que corresponde ao momento actual** cunha fotografía ou pictograma descritivo da mesma, así como a información principal da tarefa e o seu nome. Dentro desta función, o usuario tamén pode seleccionar a opción de finalización da tarefa ou a de descrición. A función de descrición **mostra unha descrición detallada da tarefa actual** dunha maneira gráfica (con números e pictogramas) para que o usuario saiba en que consiste a tarefa.

No caso de que unha tarefa finalice, xa sexa porque rematou o tempo que estaba establecido para a mesma ou porque o usuario indicou que a rematou, non se mostrarán máis tarefas ata que sexa a hora da seguinte tarefa programada.

O usuario sempre terá **acceso a unha listaxe cos pictogramas ou imaxes das tarefas restantes do día** en orde cronolóxica, as cales irán desaparecendo desta listaxe a medida que se realicen. Esta funcionalidade resulta chave para axudarlle ao neno con TEA a anticipar as actividades que virán a continuación.

O usuario en todo momento terá acceso a unha barra de progreso para ser **notificado** do tempo que leva realizando a tarefa e do tempo que lle resta para rematala. Este tipo de información non resulta intrusiva, é moi visual e minimiza a posibilidade de que o neno se disperse con ela.

7.2.3 Requisitos non funcionais

Requisitos de aparencia

Coma se mencionará na sección 8.3.1, a aparencia da aplicación ten que cumprir o estándar Material Design. Isto é debido a que a aplicación ten que cumprir un deseño minimalista e non cargado, centrándose nas imaxes e non no texto debido ao tipo de usuarios aos que está dirixida.

Requisitos de usabilidade

A sección do usuario debe poder ser utilizada por calquera persoa con algún TEA ou con algún problema de comprensión lectora. Resulta importante que as transicións entre as diferentes vistas desta parte da aplicación sexan fluídas e o tempo de resposta sexa curto, posto que os nenos con TEA tenden a angustiarse ante esperas cando manexan dispositivos electrónicos.

A parte correspondente ao administrador pode ser utilizada por calquera persoa, aínda que o usuario que a utilizará da mellor maneira posible será o educador ou educadora do usuario. Isto é así porque estas persoas saben como definir as rutinas e as tarefas destes usuarios.

Requisitos de seguridade

Unicamente poderán acceder á parte administrativa da aplicación as persoas con permisos de administrador (usuario e contrasinal).

Requisitos de hardware

Sistema operativo Android cunha versión superior á 8.0 ou iOS cunha versión superior á 12.0 no dispositivo móbil no que se utilice a aplicación.

7.3 Casos de uso

A continuación explicamos os casos de uso posibles para os actores da aplicación. Estes casos de uso cobren os requisitos descritos na sección 7.2.

7.3.1 Casos de uso do usuario non autenticado

Rexistro. Permite crear un usuario autenticado e un administrador para poder obter un acceso persoalizado á aplicación.

Na fase de rexistro, introducirase un dato identificativo do usuario, como o seu nome e, ademais, introducíranse os datos administrativos correspondentes ao administrador, que serán os seus datos de acceso. En concreto, estes datos son o correo electrónico, o contrasinal de acceso e o contrasinal de edición.

Para forzar sempre a existencia como mínimo dun administrador por cada usuario autenticado, nesta fase de rexistro tamén se definirá o primeiro administrador da conta co seu correspondente contrasinal.

Login. Permite autenticar un usuario creado previamente.

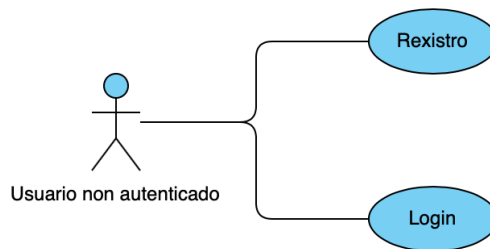


Figura 7.1: Diagrama de casos de uso correspondente ao usuario non autenticado.

7.3.2 Casos de uso do administrador

Login Administrador. Permite ao usuario da aplicación autenticarse como administrador se coñece o usuario e contrasinal de edición.

Pechar Sesión. Caso de uso que permite a un usuario con permisos de administrador pechar a súa sesión na aplicación, non estando permitido desta maneira que o usuario sen permisos de administrador poida pechar a sesión.

Mostrar Menú Administrador. Caso de uso que permite a un usuario con permisos de administrador ver o seu menú propio coas diferentes accións que pode realizar.

Cambiar Contrasinal Edición Administrador. Caso de uso que permite a un usuario con permisos de administrador cambiar o contrasinal de edición establecido nun inicio para o seu usuario administrador.

Engadir Administrador. Caso de uso que permite a un usuario con permisos de administrador engadir outro administrador. Para iso, débense introducir os datos do novo administrador que se comentaron na sección anterior. Esta é a maneira de que o novo administrador creado quede vinculado directamente ao usuario existente, compartindo os seus datos.

Ver Tarefas. Caso de uso que permite a un usuario con permisos de administrador ver todas as tarefas previamente creadas por todos os administradores. Ten a opción de buscar unha tarefa específica filtrando polo seu nome.

Engadir Tarefa. Caso de uso que permite a un usuario con permisos de administrador crear unha nova tarefa, tendo que proporcionar un nome, unha imaxe (que poderá elixir do banco de pictogramas ou ben utilizar unha propia), unha descrición que será unha listaxe de pictogramas, a súa duración e un tipo para determinar se quere que a tarefa sexa avaliada ou non.

Eliminar Tarefa. Caso de uso que permite a un usuario con permisos de administrador eliminar os datos dunha tarefa.

Ver Rutinas. Caso de uso que permite a un usuario de tipo administrador ver todas as rutinas creadas previamente por todos os usuarios de tipo administrador, podendo filtralas por nome.

Engadir Rutina. Caso de uso que permite a un usuario de tipo administrador engadir unha nova rutina ao banco de rutinas existente.

Cada rutina ten un nome e está composta de tarefas ordenadas asignadas a un momento do día. Unha rutina non pode estar composta de tarefas que impliquen un tempo de realización total superior a un día nin se poden superpoñer tarefas dentro da mesma rutina. Por exemplo, unha tarefa asignada ás 14:00 horas que dura 30 minutos non pode estar na mesma rutina que outra tarefa asignada ás 14:15 que dura 30 minutos.

Eliminar Rutina. Caso de uso que permite a un usuario con permisos de administrador eliminar os datos dunha rutina.

Modificar Rutina. Caso de uso que permite a un usuario con permisos de administrador modificar os datos dunha rutina existente, coma pode ser engadir tarefas á rutina ou desasignar tarefas da rutina.

Asignar e Desasignar Rutinas a Días. Caso de uso que permite a un usuario con permisos de administrador asignar unha rutina a un día concreto para que, posteriormente, a aplicación amose ao usuario que non é administrador as tarefas da rutina dese día. Con

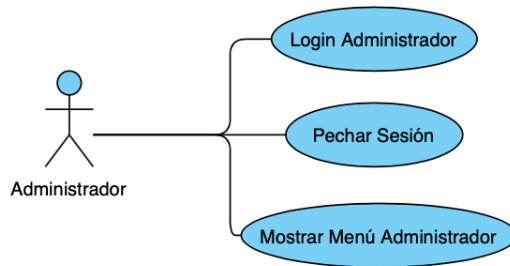


Figura 7.2: Diagrama de casos de uso correspondente ao menú do administrador.

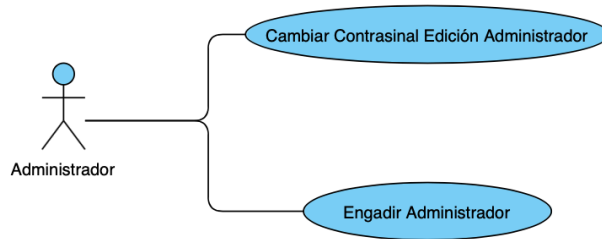


Figura 7.3: Diagrama de casos de uso correspondente á xestión de usuarios do administrador.

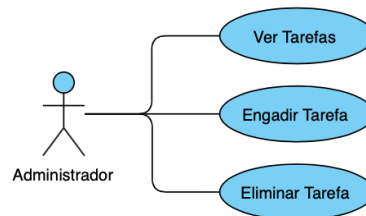


Figura 7.4: Diagrama de casos de uso correspondente á xestión de tarefas do administrador.

este caso de uso tamén se pode consultar o calendario coas rutinas asignadas, podendo do mesmo xeito desasignalas dos días. Unicamente se pode asignar unha rutina por día.

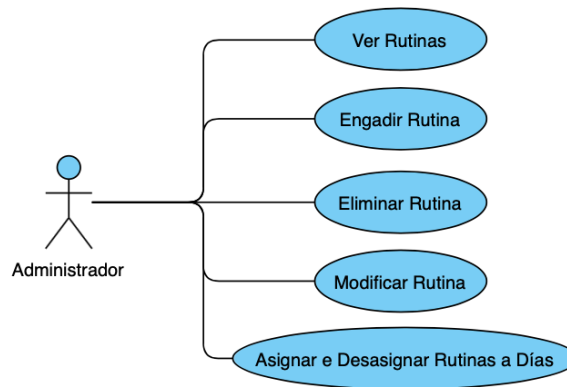


Figura 7.5: Diagrama de casos de uso correspondente á xestión de rutinas do administrador.

Ver Estatísticas. Caso de uso que permite a un usuario con permisos de administrador ver as estatísticas correspondentes do usuario sen permiso de administrador sobre as tarefas realizadas. Permítense ver as porcentaxes de tarefas realizadas correctamente e incorrectamente e de tempo utilizado para a realización de cada unha delas con respecto ao tempo total asignado.

Comparar Datos Caso de uso que permite a un usuario con permisos de administrador comparar os datos estatísticos mencionados anteriormente entre as diferentes tarefas.

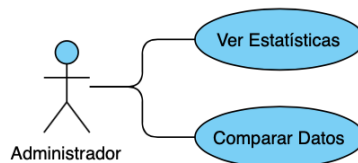


Figura 7.6: Diagrama de casos de uso correspondente á xestión de estatísticas do administrador.

7.3.3 Casos de uso para o usuario

Mostrar Tarefa Actual. A aplicación permite ver a información xeral da tarefa que corresponde realizar nese momento. No caso de que non exista ningunha tarefa a realizar nun

determinado momento, ben porque non haxa nada programado ou porque o usuario finalizou a tarefa antes do previsto, mostrarase unha tarefa xenérica ou de “recompensa”. A información xeral que se amosa para unha tarefa é a imaxe/pictograma asociada á mesma, o seu nome e a duración correspondente.

Finalizar Tarefa Actual. O usuario pode dar por finalizada a tarefa actual cando transcorre unha porcentaxe de tempo determinada con respecto á duración total da tarefa.

Avaliar Tarefa Actual. Ao finalizar unha tarefa que teña como tipo Avaliar o usuario/coi-dador indicará se conseguiu realizar a tarefa correctamente ou non.

Mostrar Descrición da Tarefa Actual. Este caso de uso permite amosar máis información sobre a tarefa actual. Esta información consistirá nunha descrición gráfica da tarefa dividida en diferentes subtarefas entendibles para o usuario.

Mostrar Tarefas Restantes Mostra unha sucesión de imaxes/pictogramas coas seguintes tarefas ordenadas que compoñen a rutina que se está levando a cabo.

Cambiar Cor Aplicación Caso de uso que permite a un usuario cambiar a cor e a aparencia xeral de toda a aplicación.

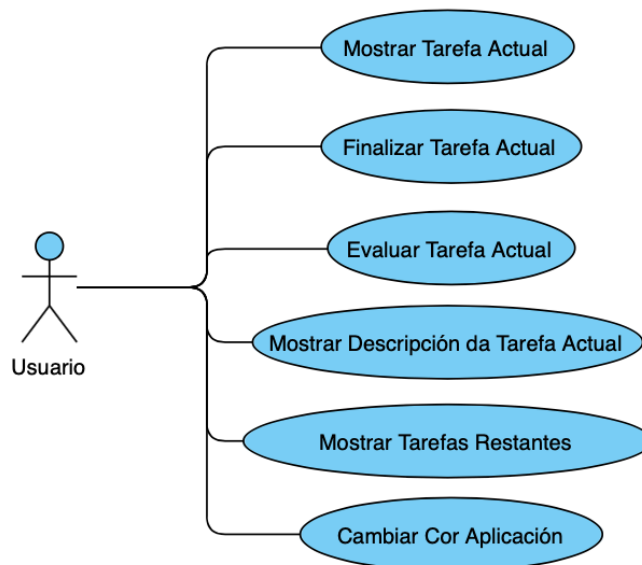


Figura 7.7: Diagrama de casos de uso correspondente ao usuario.

7.4 Fluxo da aplicación

A secuencia de todas as posibles accións que se poden realizar con esta aplicación é definido polo diagrama de fluxo representado na figura 7.8. Este diagrama de fluxo está composto por accións, condicións e transicións. Nel pódense observar como realizar todas as posibles accións que permite a aplicación e as consecuencias desas accións.

7.5 Esquema de Datos da Aplicación

O esquema de datos non relacional que utiliza a aplicación está definido polo diagrama da figura 7.9. Como se pode apreciar, esta figura representa a información sobre as diferentes coleccións de datos a utilizar na aplicación así como os seus atributos e os seus tipos.

Firebase, a diferencia das bases de datos relacionais, almacena os datos nun formato estruturado da mesma maneira que fai JSON.

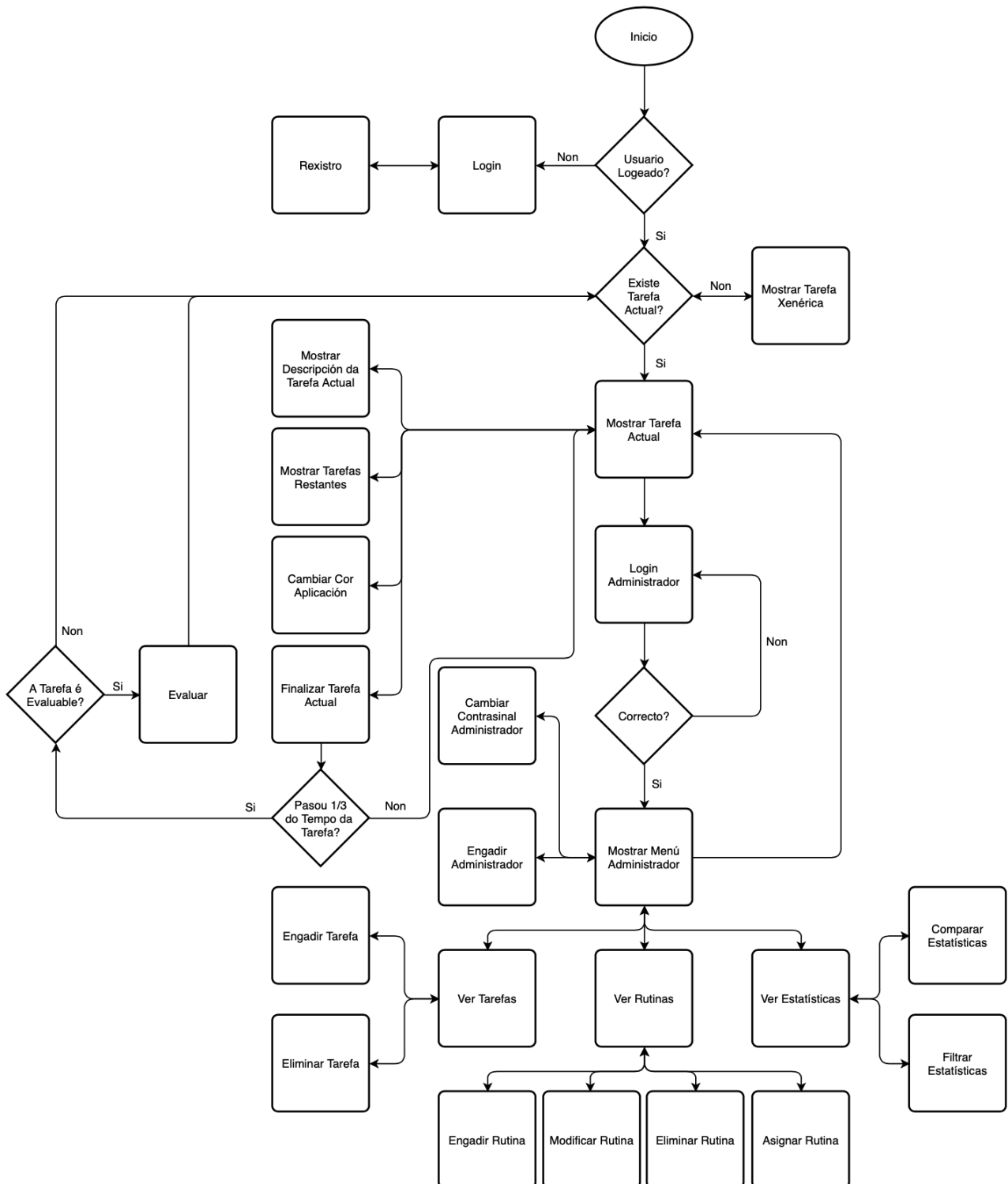


Figura 7.8: Diagrama de flujo da aplicación.

```

{
  "administradores":{
    "idAdmin":<String>{
      "nombre":<String>,
      "mail":<String>,
      "password":<String>,
      "usuario":<String>"
    }
  }
}
{
  "usuarios":{
    "idUsuario":<String>{
      "nombre":<String>,
      "tareass":{
        "idTarea":<String>{
          "nombre":<String>,
          "descripcion":<List<String>>,
          "duracion":<String>,
          "creador":<String>,
          "pictograma":<String>,
          "tipo":<String>,
          "uploaded":<bool>"
        }
      }
    }
  },
  "rutinas":{
    "idRutina":<String>{
      "nombre":<String>,
      "tareass":<String>{
        "fin":<DateTime>,
        "inicio":<DateTime>,
        "tarea":<String>,
      },
      "creador":<String>,
    }
  },
  "calendario":{
    "<DateTime>":{"idRutina":<String>},
  },
  "estadísticas":<String>{
    "evaluada":idTarea<String>{
      "idEvaluada":<String>{
        "evaluada":<bool>,
      }
    },
    "porcentajeTiempo":idTarea<String>{
      "idPorcentaje":<String>{
        "porcentaje":<double>,
      }
    }
  }
}
}
}
}

```

Figura 7.9: Modelo JSON que representa a base de datos non relacional.

Capítulo 8

Deseño

NESTE capítulo vaise describir o patrón de deseño utilizado para a aplicación, o deseño lóxico da base de datos que se utilizará e o estándar para o deseño da vista utilizado. Tamén se inclúen os bosquejos iniciais do deseño da interface de usuario da aplicación.

8.1 Patrón de deseño

O patrón de deseño que segue a aplicación é o Modelo Vista Controlador (MVC). Este patrón de deseño é común no desenvolvemento de software no que intervéñ unha interface, xa que diferencia a implementación desta interface de usuario dos mecanismos necesarios para obter os datos que se lle mostran ao usuario.

Modelo. O modelo é o compoñente principal dentro deste patrón. Xestiona os datos, regras e lóxica a nivel de aplicación independentemente da vista. Inclúe a implementación das funcionalidades, e o procesado e almacenamento dos datos do sistema.

Vista. A vista é o compoñente encargado de presentar a información proporcionada polo modelo ao usuario final. Pode representar dita información total ou parcialmente, e a mesma información pode ser representada de diferentes maneiras pola vista.

Controlador. O controlador é o compoñente de deseño encargado de comunicar a vista co modelo. Deste xeito, un cambio na implementación da vista non ten por que afectar ao modelo e viceversa.

Na figura 8.1 está representado coma interactúan os diferentes compoñentes deste patrón e o usuario.

8.2 Modelo lóxico da base de datos

Como se explicará máis detalladamente no capítulo de implementación, finalmente optouse pola realización da aplicación en castelán, polo que os nomes das diferentes coleccións e dos seus datos correspondentes están representados neste capítulo como figuran na base de datos, en castelán e en inglés.

A base de datos pode ser representada como un modelo lóxico, obtendo así un maior detalle dos contidos das coleccións e unha definición dos seus datos.

8.2.1 Colección Administradores

A colección *Administradores* serve para almacenar a información dos diferentes administradores do sistema. Cada administrador terá un identificador único que se utiliza a nivel da implementación e non se lle amosa ao usuario, un nome, un email, un contrasinal de edición e un identificador do usuario asociado, que será o neno que está a titorizar.

Admin		
Nome	Tipo de Dato	Descrición
idAdmin	String	<i>Chave primaria da colección</i> Identificador do administrador
nombre	String	Nome asociado ao administrador
mail	String	Email de contacto do administrador
password	String	Contrasinal de edición correspondente ao administrador
usuario	String	Identificador do usuario que está a supervisar este administrador

Táboa 8.1: Colección correspondente aos datos do/s administrador/es.

8.2.2 Colección Usuarios

A colección *Usuarios* serve para almacenar a información dos diferentes usuarios non administradores, é dicir, dos nenos con TEA. Cada usuario terá un identificador único que se utiliza ao nivel da implementación e non se lle amosa ao usuario, un nome e catro subcoleccións correspondentes aos seus datos, sendo estas subcoleccións Tareas, Rutinas e Estadísticas.

8.2.3 Subcolección Tareas

A colección *Tareas* serve para almacenar toda a información que se mostrará sobre as diferentes tarefas creadas.

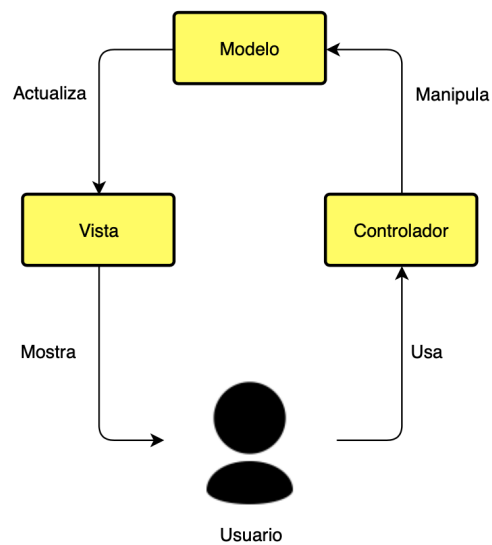


Figura 8.1: Modelo conceptual do patrón Modelo Vista Controlador.

Usuarios		
Nome	Tipo de Dato	Descrición
idUsuario	String	<i>Chave primaria da colección</i> Identificador do usuario
nombre	String	Nome asociado ao usuario
tareas	Colección	Subcolección correspondente ás tarefas dispoñibles para o usuario
rutinas	Colección	Subcolección correspondente ás rutinas do usuario
calendario	[DateTime : idRutina]	Información correspondente á asignación de rutinas a días
estadísticas	Colección	Subcolección correspondente ás estadísticas de uso do usuario

Táboa 8.2: Colección correspondente aos datos do/s usuario/s.

Tareas		
Nome	Tipo de Dato	Descrición
idTarea	String	<i>Chave primaria da colección</i> Identificador da tarefa
nombre	String	Nome asociado á tarefa
descripcion	List<String>	Listaxe ordenada de pictogramas asociados á descrición da tarefa
duración	String	Duración en formato HH:mm da tarefa
creador	String	Nome do administrador que creou a tarefa
pictograma	String	Pictograma asociada á tarefa
tipo	String	Tipo da tarefa, pode ser avaliada e ou non avaliada
uploaded	Boolean	Indicador de se a tarefa está respresentada cunha imaxe propia ou non

Táboa 8.3: Subcolección correspondente aos datos da/s tarefa/s.

8.2.4 Subcolección Rutinas

A colección *Rutinas* é a encargada de almacenar os datos das rutinas e, da mesma maneira que ocorre na colección *Tareas*, o campo creador é o nome do administrador que creou esta rutina. As rutinas están formadas por tarefas, polo que esta colección contén un campo con outra subcolección que corresponde ás tarefas das que está composta a rutina e as súas correspondentes horas de inicio e fin.

Rutinas		
Nome	Tipo de Dato	Descrición
idRutina	String	<i>Chave primaria da colección</i> Identificador da rutina
nombre	String	Nome asociado á rutina
tareas	Colección	Listaxe de tarefas e a súa hora de comezo e fin asignadas que compoñen a rutina
creador	String	Nome do administrador que creou a rutina

Táboa 8.4: Subcolección correspondente aos datos da/s rutina/s.

8.2.5 Subcolección Tareas de Rutinas

A colección *Tareas de Rutinas* é a encargada de almacenar os datos de en que momento, dentro das 24 horas que pode durar a rutina, hai que realizar cada tarefa. Desta maneira, esta colección ten uns campos inicio e fin que indicarán o momento no que a tarefa co identificador almacenado en idTarefa ten que transcorrer dentro da rutina.

Tareas de Rutina		
Nome	Tipo de Dato	Descrición
tareas	String	<i>Chave primaria da colección</i> Identificador da tarefa dentro da rutina
tarea	String	Id asociado á tarefa correspondente
inicio	DateTime	Momento correspondente ao inicio da tarefa dentro da rutina
fin	DateTime	Momento correspondente ao fin da tarefa dentro da rutina

Táboa 8.5: Subcolección correspondente aos datos da/s tarefa/s dentro da/s rutina/s.

8.2.6 Subcolección Estadísticas

A colección *Estadísticas* é a encargada de almacenar para cada realización dunha tarefa os seus correspondentes valores para, logo, poder formar as estatísticas correspondentes. Os valores que se almacenan son a correcta ou non correcta realización da mesma e a porcentaxe do tempo dispoñible utilizado para a súa realización.

Cada vez que se gardan datos sobre unha determinada tarefa, créase un novo `idEvaluada` co identificador desa tarefa e almacénanse os datos correspondentes.

Estadísticas		
Nome	Tipo de Dato	Descrición
<code>idEvaluada</code>	String	Identificador da avaliación
<code>evaluada</code>	<code>[idTarea:idEvaluada]:boolean</code>	Indica se a tarefa se realizou correctamente ou non
<code>idPorcentaje</code>	String	Identificador da porcentaxe
<code>porcentajeTiempo</code>	<code>[idTarea:idPorcentaje]:double</code>	Porcentaxe do tempo dispoñible utilizado

Táboa 8.6: Subcolección correspondente aos datos da/s tarefa/s dentro da/s rutina/s.

8.3 Deseño da interface de usuario

A interface de usuario que segue a aplicación é unha interface cun deseño moi sinxelo para o usuario sen permisos de administrador, focalizada nas imaxes e na accesibilidade, simplicidade e facilidade de uso das opcións de usuario. Isto vén motivado polo tipo de usuario ao que está destinada esta parte da aplicación.

Por outra banda, a interface da aplicación para o usuario con permisos de administrador está orientada á optimización da organización e administración das tarefas, rutinas e pictogramas dispoñibles. Por esa razón, priorízase a funcionalidade respecto da vistosidade ou da sinxeleza debido, entre outras cousas, á carga de listaxes ou galerías que se mostran agora en lugar dunha única imaxe, como ocorría no caso do usuario non administrador.

Para a distribución dos elementos de navegación dentro da aplicación e a selección das paletas de cores e das animacións úsase o estándar Material Design, que se explicará no seguinte apartado.

8.3.1 Material Design

Material Design é un estándar de deseño creado por Google que se centra na parte de visualización para sistemas operativos Android, desde a súa versión Android Lollipop (Android 5.0).

Os elementos máis identificativos de Material Design son: seguimento dunhas liñas e deseño xeral sinxelos, minimización da sobrecarga de elementos do deseño, apostando por compoñentes sinxelos como pode ser un botón, notificacións en forma de tarxetas sobre a aplicación ou a inclusión de botóns flotantes sobre a mesma.

8.3.2 Bosquexos iniciais de deseño da interface de usuario

Nesta sección móstranse os bosquexos ou *mockups* previos á implementación da aplicación.

Os bosquexos representados nas figuras 8.2 e 8.3, amosan como sería a primeira interacción do usuario coa aplicación ao non estar logueado previamente. Desta maneira, co coñecemento dunhas credenciais concretas o usuario pode iniciar sesión para ter o seu contido personalizado ou pode rexistrarse como un novo usuario para definir así un usuario e unhas novas credenciais propias.



Figura 8.2: Bosquexo correspondente ao rexistro inicial do usuario.



Figura 8.3: Bosquexo correspondente ao login inicial do usuario.

O bosquexo representado na figura 8.4 mostra como sería a interacción do usuario non administrador coa aplicación para visualizar a tarefa actual e as tarefas restantes.

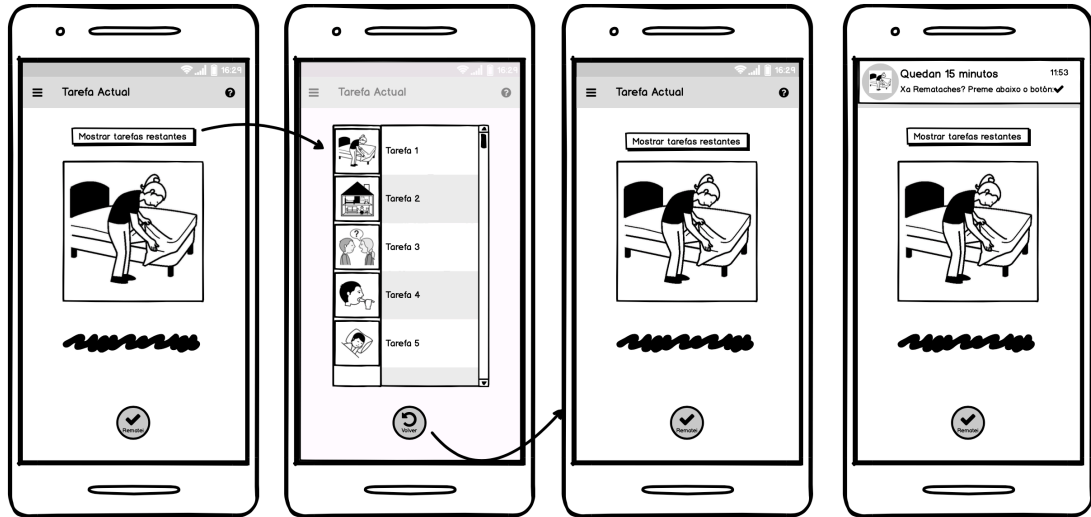


Figura 8.4: Bosquexo correspondente a mostrar tarefas restantes e coma funcionarían as notificacións da aplicación.

O bosquexo representado na figura 8.5 amosa como sería a interacción ata chegar ao menú do administrador. É importante destacar que o feito de pedir un contrasinal impide que un usuario que non teña permisos de administrador poida acceder á sección de administración da aplicación. Como se pode apreciar na figura, o acceso a este menú faise a través dunha opción de login situada nun menú despregable na parte esquerda da aplicación. Desta maneira, minimízase o acceso erróneo do usuario sen permisos de administrador ao menú de login do administrador. Aínda así, pode ocorrer o caso dun acceso imprevisto do usuario da aplicación (o neno con TEA) ata este menú. Por iso se decidiu colocar un gran botón representativo neste menú coa opción de retroceder, o cal permite volver á vista da tarefa actual.

O bosquexo representado na figura 8.6 amosa como acceder ao banco de rutinas e de tarefas desde o menú do administrador, e como se deberían visualizar os elementos destes bancos. Tamén se pode observar o comportamento que debería ter o botón integrado na vista do banco de rutinas, proporcionando deste xeito nun único botón dúas opcións e mantendo o deseño Material Design.

Os dous bancos teñen un buscador para poder filtrar por nome dentro das tarefas ou rutinas, respectivamente, obtendo así unicamente os resultados filtrados.

Como se pode observar, na barra de estado da aplicación en todo momento se mostrará en que pantalla nos atopamos e un botón para volver ao menú do administrador. Loxicamente, cando a aplicación está no menú do administrador non terá este botón para volver.

O bosquexo representado na figura 8.7 amosa o proceso de como se eliminaría unha tarefa

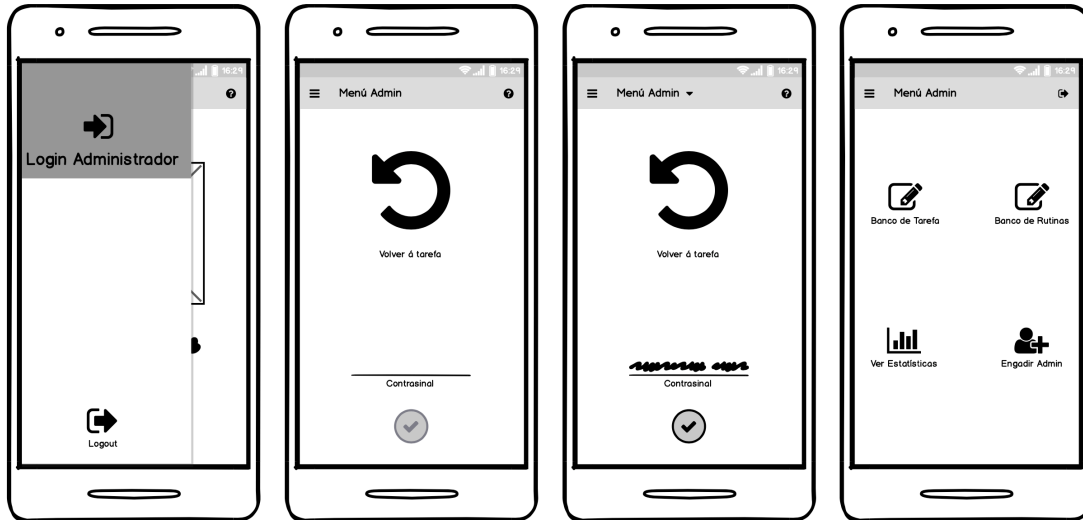


Figura 8.5: Bosquexo correspondente á acción de autenticación do administrador.

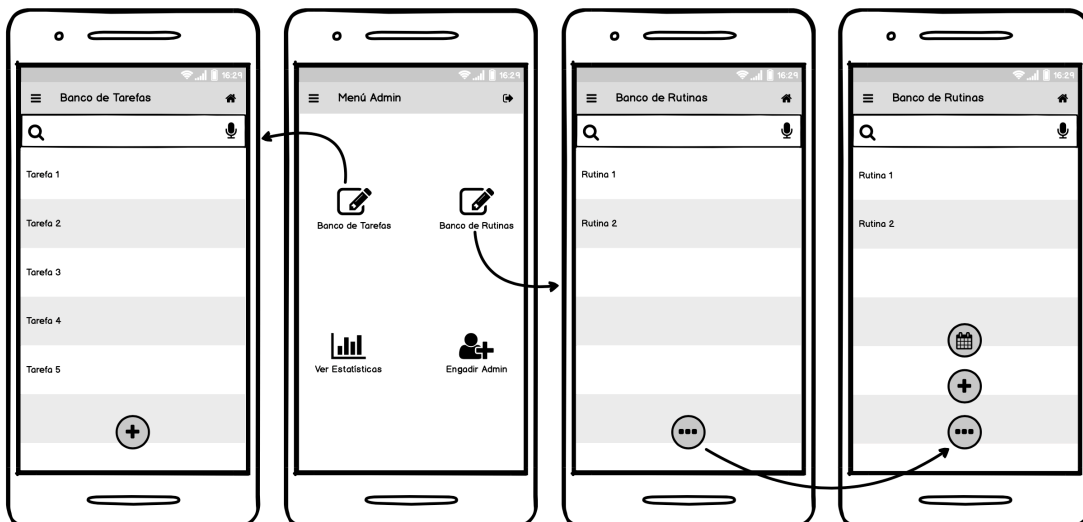


Figura 8.6: Bosquexo correspondente ao banco de tarefas e ao banco de rutinas.

do banco de tarefas. A maneira de eliminar unha rutina do banco de rutinas sería a mesma.

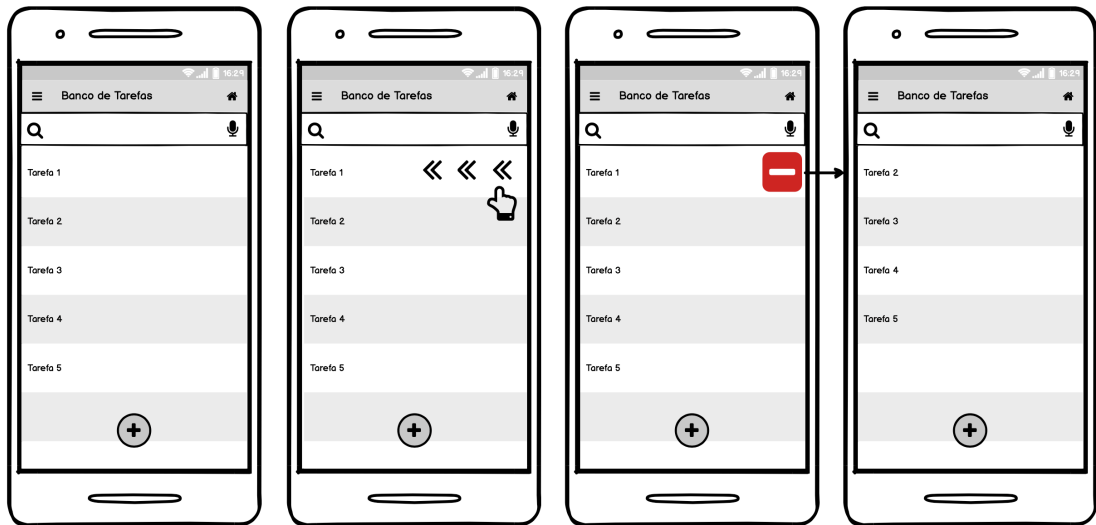


Figura 8.7: Bosquexo correspondente á eliminación dunha tarefa do banco de tarefas.

O bosquejo representado na figura 8.8 amosa o proceso de como engadir unha nova tarefa ao banco de tarefas. A nivel de interface de usuario, o proceso para a modificación de tarefas é similar. Pódese observar que o botón para finalizar este proceso de engadir tarefas está deshabilitado ata que se introducen todos os datos da tarefa e se selecciona unha imaxe.

Para acceder á vista da selección de pictogramas hai que seleccionar a icona coa imaxe baleira. Unha vez dentro desta vista, aparece como primeira opción a utilización da cámara, que abrirá a opción nativa do sistema operativo para a utilización da propia cámara ou da galería de imaxes. Por outra banda, na barra superior da aplicación, a icona para volver ao menú do administrador cambia por un botón de procura que permite filtrar pictogramas.

O bosquejo representado na figura 8.9 mostra o proceso de como engadir unha nova rutina ao banco de rutinas. A nivel de interface de usuario, o proceso para a modificación de rutinas é similar. Como se pode apreciar, para engadir unha tarefa á rutina que se vai crear, hai que seleccionar un dos *slots* baleiros e engadir unha tarefa que exista previamente no banco de tarefas. Na vista de tarefas dispoñibles indicárase o seu nome e a súa duración, sen entrar noutros detalles da tarefa. Unha vez elixida a tarefa que se desexa engadir, haberá que seleccionar unha hora de comezo da mesma que non estea ocupada xa por outras tarefas e, dese xeito, finalizaríase o proceso de engadir unha tarefa á rutina. Para eliminar tarefas da rutina realizarase da mesma maneira que na figura 8.7 para a eliminación dun elemento do seu correspondente banco.

O bosquejo representado na figura 8.10 amosa a maneira de presentar as diferentes estatísticas obtidas das tarefas, aínda que esta vista pode variar despois na fase de implementación dependendo do tipo de estatística concreta que se queira visualizar.

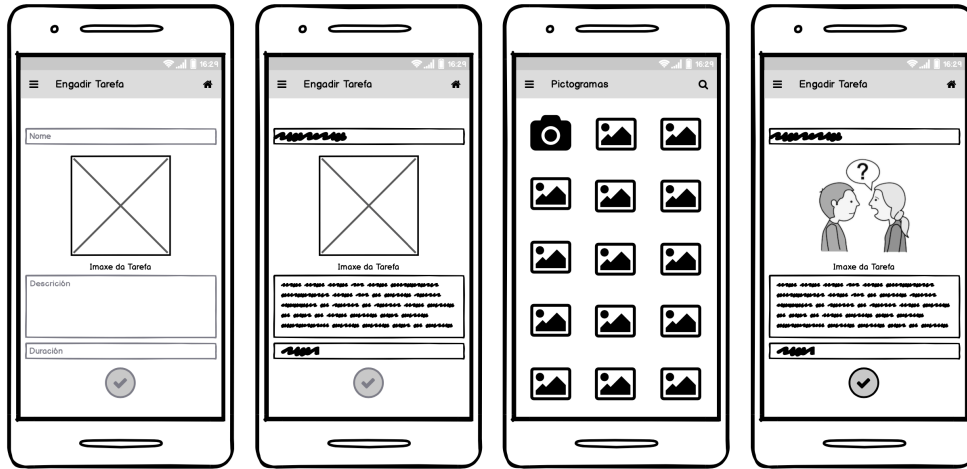


Figura 8.8: Bosquexo correspondente a engadir tarefas.



Figura 8.9: Bosquexo correspondente a engadir rutina.

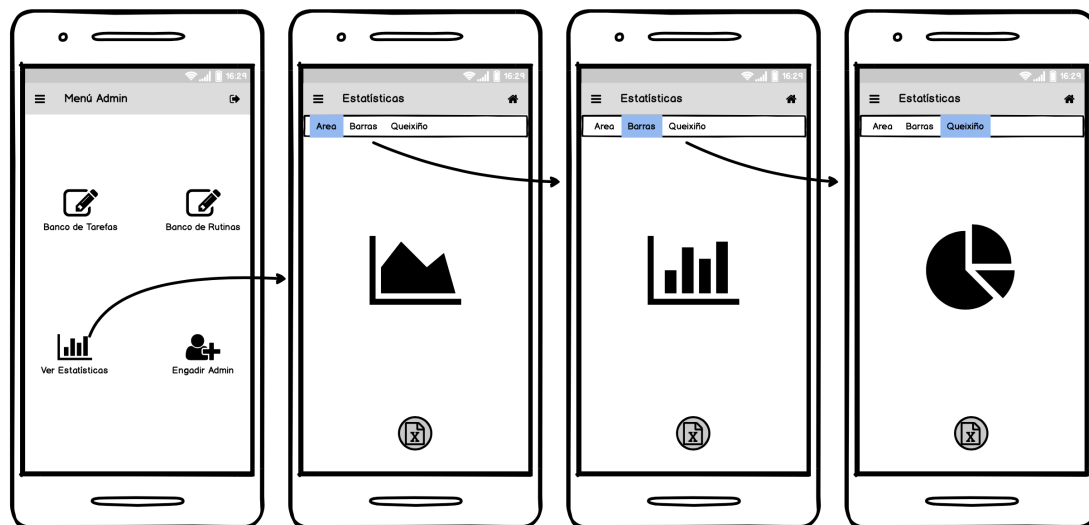


Figura 8.10: Bosquexo correspondente á visualización dos diferentes tipos de estatísticas.

O bosquexo representado na figura 8.11 ilustra a maneira na que se engaden novos usuarios con permisos de administrador desde o menú dun usuario autenticado como administrador.

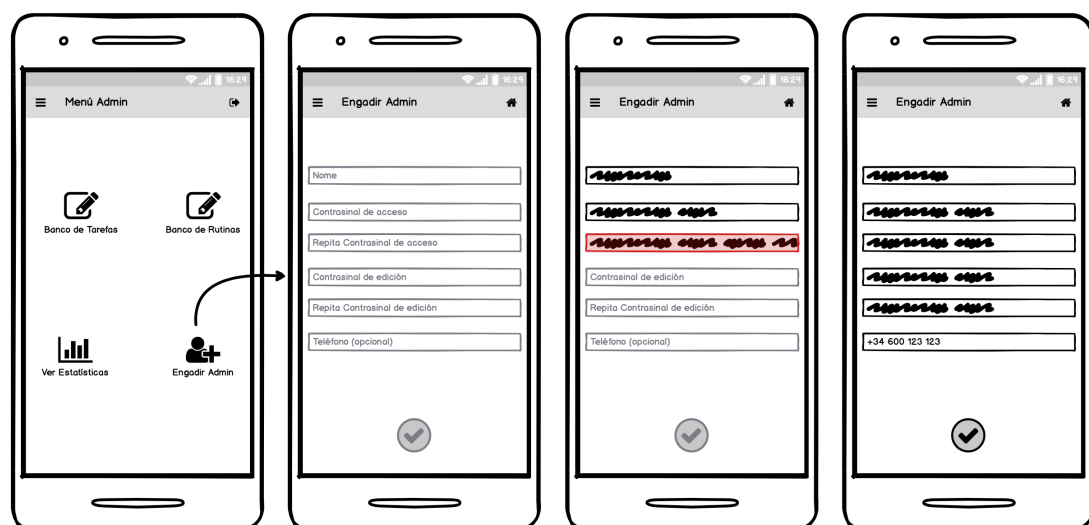


Figura 8.11: Bosquexo correspondente a engadir novos administradores.

8.4 Idioma da aplicación

Logo de estudar se implementar a aplicación en galego ou en castelán, finalmente tomouse a decisión de deseño da aplicación en castelán co obxectivo de chegar a un maior número de

persoas e impedir así que a linguaxe sexa unha barreira á hora da súa utilización.

8.5 Estrutura do proxecto

Na figura 8.12, pódese apreciar a organización ou deseño en paquetes que se decidiu para este proxecto. Como se pode apreciar, cada paquete contén o conxunto de directorios/ficheiros necesarios para a implementación dunha ou de varias das funcionalidades citadas no capítulo anterior. Os paquetes seguen unha orde en profundidade xerárquica con respecto a ditas funcionalidades para un uso máis cómodo dos mesmos.

Exceptuando o arquivo *main.dart*, todos os demais arquivos están albergados en subpaquetes *pages* ou *controller* mantendo así o patrón de deseño MVC. Desta maneira, dentro do MVC, o rol do modelo é levado a cabo por Firebase [35] e os seus módulos (Google Cloud Storage [37], Cloud Firestore [36] e Firebase Authentication [38]), o rol da vista é levado a cabo por todos os arquivos dentro dos subpaquetes *pages* e o rol de controlador é levado a cabo polos arquivos dentro dos subpaquetes *controller* que se encargan de xestionar as conexións e manexar os datos proporcionados e obtidos do modelo.

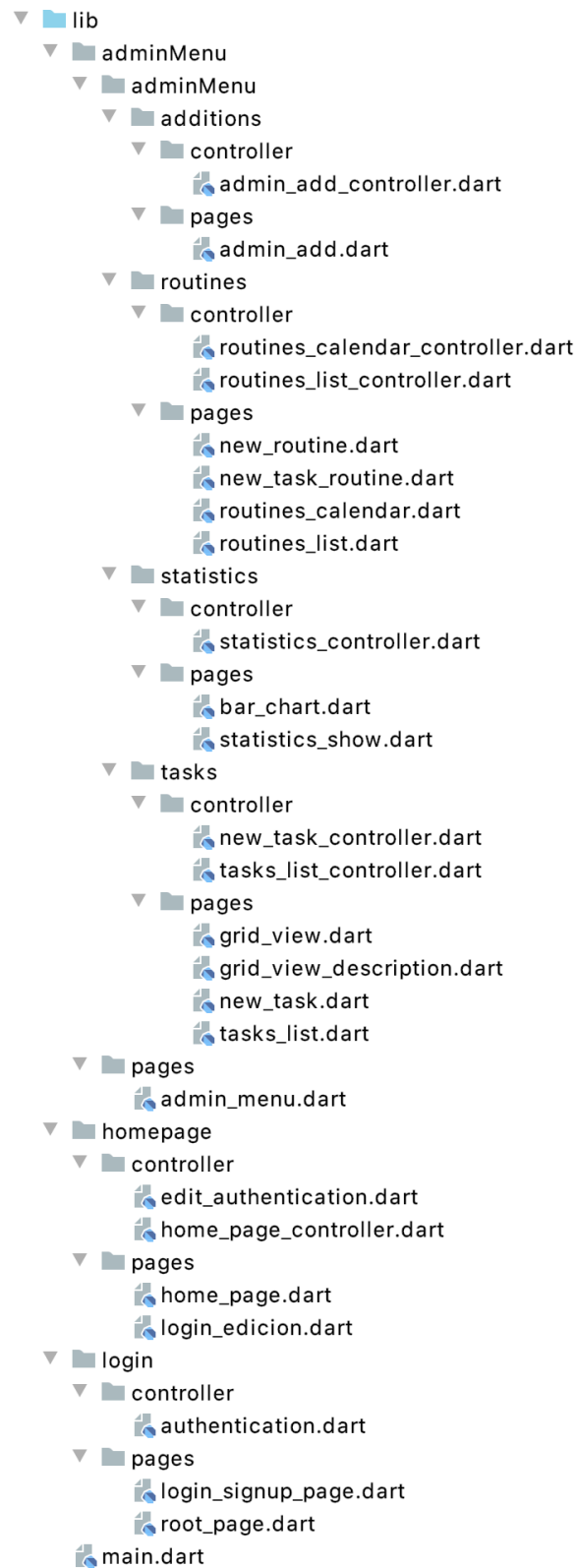


Figura 8.12: Representación en árbol de la organización de código de proyecto

Implementación

NESTE capítulo trátaranse os temas máis relevantes e destacables que xurdiron durante as diferentes fases de implementación.

9.1 Widgets para as vistas

Para seguir o estándar de deseño Material Design explicando en [8.3.1](#), farase uso da librería *material.dart* propia de Flutter.

9.1.1 Composición de widgets

Calquera elemento das vistas é un *widget*, en concreto un *widget* chamado *Scaffold*, que é o encargado de poñer unha barra superior co nome da pantalla actual, un botón flotante optativo e un corpo ou contido que está, á súa, vez composto de máis *widgets*, os cales poden estar compostos de máis *widgets*, e así recursivamente ata chegar aos *widgets* básicos ofrecidos pola librería de *material.dart*.

Un exemplo desta composición de *widgets* dentro da aplicación pode verse na figura [9.1](#), onde se mostra a pantalla da aplicación correspondente á listaxe de tarefas. Moitos destes *widgets* son empregados en diferentes vistas da aplicación polo que, gracias ao uso de Flutter, poden ser reutilizados de forma sinxela e eficiente.

9.2 Implementación dos controladores para os diferentes módulos de Firebase

Como se explicaba no capítulo de deseño, vaise utilizar Firebase e algúns dos seus módulos para aloxar e acceder ao modelo da aplicación. Para conectar estes módulos, correspondentes ao modelo, coa vista da aplicación implementáronse diferentes controladores dependendo do módulo a utilizar.

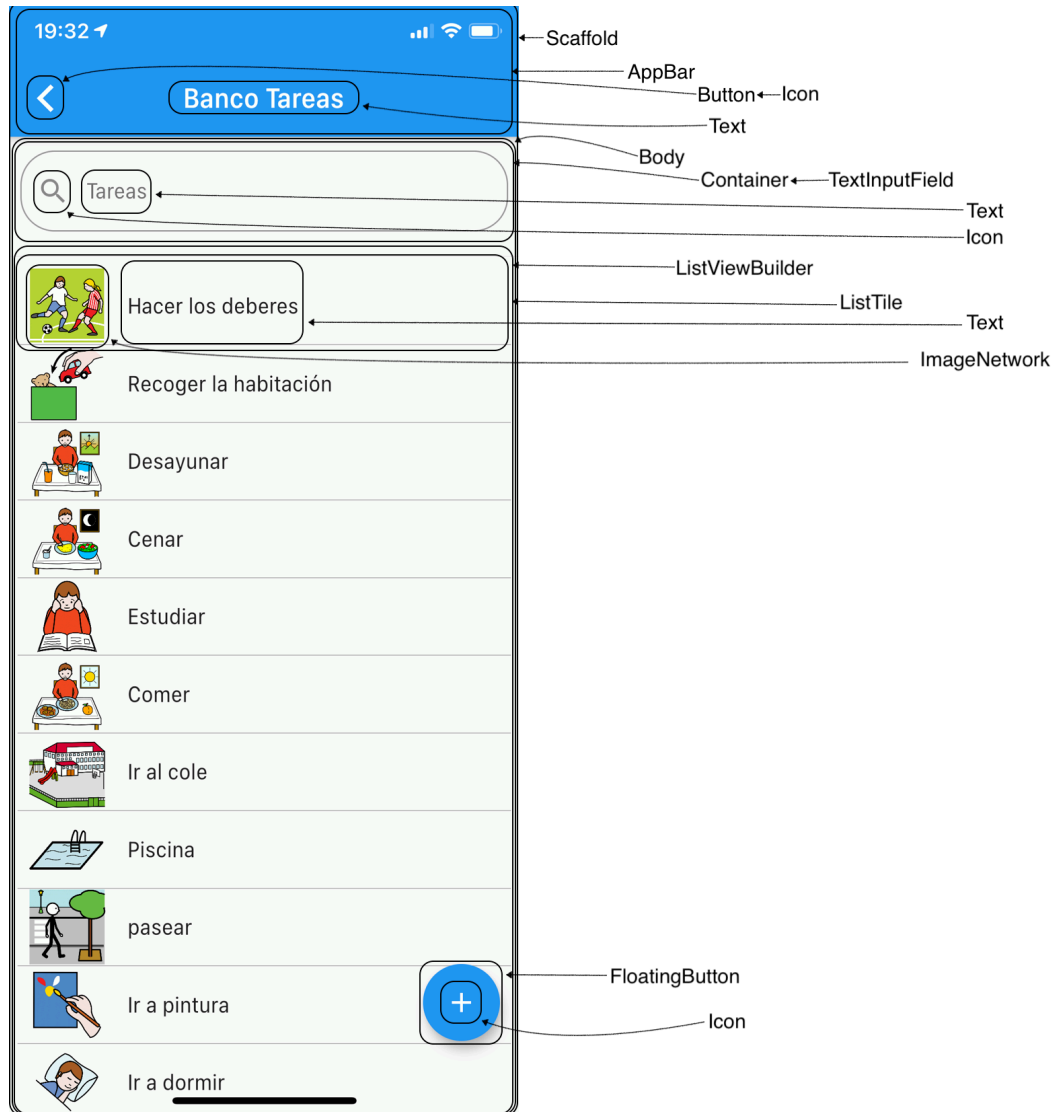


Figura 9.1: Captura explicativa da composición de widgets na listaxe das tarefas.

Nesta sección, explicaranse unha serie de exemplos de funcións que implementan as operacións dos controladores dependendo do modelo do que é necesario obter datos e do módulo de Firebase empregado.

- No caso dos controladores encargados de xestionar o manexo dos datos relacionados coa xestión de usuarios (Firebase Authentication), utilízanse funcións moi básicas baseadas en funcións xa existentes grazas á librería de Firebase Authentication (*firebase_auth.dart*).

Un exemplo dentro da aplicación pode ser o seguinte código, que corresponde á función encargada de dar de alta un novo usuario cun correo electrónico e un contrasinal, e que fai uso da función *createUserWithEmailAndPassword(email, password)* proporcionada por Firebase.

```
1 Future<String> signUp(String email, String password) async{
2     AuthResult result = await
      _firebaseAuth.createUserWithEmailAndPassword(email: email,
      password: password);
3     FirebaseUser user = result.user;
4     return user.uid;
5 }
```

- No caso da conexión coa base de datos de Cloud Firestore o proceso cambia, xa que non existen funcións predefinidas tan elaboradas como pode ser no caso de Firebase Authentication. Debido a isto, é necesario definir funcións máis elaboradas e cunha finalidade máis concreta para o uso dentro do programa.

Un exemplo dentro da aplicación pode ser o seguinte código, que é a función encargada de obter as tarefas que forman parte dunha rutina a partires do seu identificador.

```
1 Future<dynamic> getRoutineData(firestoreInstance, auth, id) async {
2     try {
3         FirebaseUser _user = await auth.getCurrentUser();
4         String _userId = _user.uid;
5         var data = await firestoreInstance
6             .collection('administradores')
7             .document(_userId)
8             .get();
9         String _kidId = data.data['usuario'];
10        var tasks = await firestoreInstance
11            .collection("usuarios")
12            .document(_kidId)
13            .collection('rutinas')
14            .document(id)
15            .collection('tarefas')
```

```

16         .snapshots();
17     return (tasks);
18 } catch (e) {
19     print('Error: $e');
20 }
21 }

```

- Cando se engaden novos datos á base de datos, os identificadores dos mesmos poden xerarse manualmente (á hora de implementar decídese que identificador vai ter cada dato) ou automaticamente (permitindo a Firebase xerar os identificadores). Na implementación, utilízase en case todo momento a xeración de identificadores automática por comodidade á hora da implementación, excepto no caso do calendario. Neste caso, debido a que cada día é único, pode utilizarse o propio *DateTime* do día como identificador.
- Finalmente, no caso da xestión de Google Cloud Storage, que se utiliza para almacenar as fotografías propias que decida o administrador, utilízase a librería *firebase_storage.dart* para almacenar as fotografías e para xerar unha dirección URL que dea acceso a cada unha desas fotografías. Esa dirección, posteriormente, será almacenada na base de datos de Cloud Firestore como a dirección URL do pictograma asignado á tarefa que se está creando.

Un exemplo deste proceso pode verse nunha sección do código correspondente á función encargada de engadir as tarefas á base de datos. Esta sección do código é a seguinte:

```

1 String _imageId = Uuid().v1();
2 String imagePath = "$_imageId.png";
3 final StorageReference storageReference =
4   FirebaseStorage().ref().child(imagePath);
5 final StorageUploadTask uploadTask =
6   storageReference.putFile(image);
7 await uploadTask.onComplete;
8 uploaded = true;
9 pictoUrl = await storageReference.getDownloadURL();

```

9.3 API ARASAAC

Un dos aspectos chave na aplicación é o uso de pictogramas, tanto para especificarlle visualmente ao neno o traballo a realizar en cada unha das tarefas como para permitir unha descrición destas en subtarefas máis sinxelas. Como se comentou no capítulo 4, os pictogramas que adoitan empregar moitos dos profesionais para definir as rutinas, son os pictogramas

de ARASAAC. Para a inclusión destes pictogramas na aplicación utilízouse a API que esta asociación ofrece. Aínda que esta API está nunha versión moi temperá (0.0.1), ofrece un servizo suficiente para poder implementar as funcións necesarias nesta aplicación que requiran o uso de pictogramas.

Para evitar un almacenamento excesivo tanto na base de datos coma na memoria interna do dispositivo móbil, realízanse peticións HTTP para a obtención dos pictogramas en liña, que teñen un formato de imaxe con extensión .png. Desta maneira, ARASAAC é o encargado de albergalos no seu servidor evitando o uso do espazo tanto local do dispositivo móbil do usuario como do espazo habilitado de Firebase.

Un exemplo de uso desta API dentro da aplicación é no buscador interactivo de pictogramas á hora de crear unha nova tarefa. Este sería o código correspondente a este exemplo:

```
1 Future<List<String>> getSearch(busqueda) async {  
2   List<String> myListIds = List<String>();  
3   if (busqueda != "" && busqueda != null) {  
4     var pictogram = jsonDecode((await http.get(  
5  
6       'https://api.arasaac.org/api/pictograms/es/search/$busqueda'))  
7       .body);  
8     for (int i = 0; i < pictogram.length; i++) {  
9       myListIds.add(pictogram[i]['_id'].toString());  
10    }  
11  } else {  
12    await Future.delayed(Duration(  
13      milliseconds: 500)); //Simulate connection and make the  
14    setstate  
15  }  
16  return myListIds;  
17 }
```

A figura 9.2 amosa unha captura do resultado deste tipo de procuras de pictogramas na aplicación real correndo nun dispositivo.

9.4 Utilizar imaxes propias do dispositivo móbil

Á hora de escoller unha imaxe para unha tarefa pódese escoller entre os pictogramas obtidos da API de ARASAAC ou existe a posibilidade de utilizar directamente unha imaxe proporcionada polo dispositivo.

O resultado da implementación desta funcionalidade pode verse nas dúas primeiras opcións da figura 9.2. Neste caso, a imaxe que se utiliza para unha nova tarefa pode ser unha das imaxes xa existentes no almacenamento interno do dispositivo móbil ou calquera imaxe obtida dunha foto coa aplicación nativa da cámara do dispositivo.

Esta é a única funcionalidade da aplicación na que hai que realizar diferentes accións dependendo de se a aplicación se está a executar nun dispositivo con sistema operativo iOS ou Android. Isto é debido a que a maneira de xestionar as peticións nativas de permisos para acceder aos arquivos, é dicir, a maneira de acceder á cámara do sistema operativo e de obter o arquivo correspondente á imaxe, é diferente dependendo do sistema operativo no que se estea a executar a aplicación.

A maneira de implementar esta petición de permisos en iOS é modificando o arquivo `/ios/Runner/Info.plist`. No seguinte cadro, achéganse os permisos solicitados ao sistema:

```

1 <key>NSPhotoLibraryUsageDescription</key>
2 <string>Esta aplicación requiere acceso a la galería</string>
3 <key>NSCameraUsageDescription</key>
4 <string>Esta aplicación requiere acceso a la cámara</string>

```

A maneira de implementar a solicitude de permisos en Android está soportada de maneira nativa a partires de Android 10. Pola contra, cunha versión inferior hai que engadir o atributo `android:requestLegacyExternalStorage="true"` na sección `<application>` do arquivo `AndroidManifest.xml`.

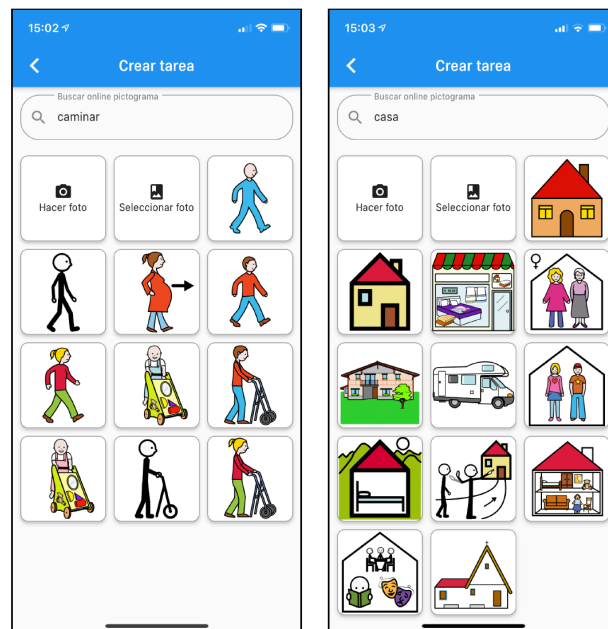


Figura 9.2: Capturas da procura de pictogramas correndo na aplicación real.

9.5 Implementación das rutinas

9.5.1 Cambios respecto ao deseño inicial

O deseño final desta parte da aplicación, comparado co do bosquexo inicial que se pode consultar na figura 8.9 do capítulo de deseño, sufriu pequenas variacións.

Para amosar as tarefas que compoñen unha rutina, sen perder o formato de listaxe nin ningún tipo de información, deuse unha maior prioridade aos pictogramas de cada tarefa na listaxe das tarefas. Estes pictogramas ademais quedan unidos por unha liña de tempo, todo isto co obxectivo de obter un resultado máis visual na propia listaxe. Isto cobra especial relevancia no caso da vista onde o neno con TEA pode visualizar o resto de tarefas que ten programadas a continuación na súa rutina.

Outro cambio sobre o deseño inicial foi o reloxo para seleccionar a hora, implementándose un reloxo máis elaborado e interactivo que cambia dependendo da hora do día que se seleccione. Esta implementación tratouse de facer o máis visual e intuitiva posible para o usuario administrador, mellorando a facilidade de uso da aplicación e mantendo na medida do posible o deseño inicial.

Todos estes cambios poden apreciarse na figura 9.3.

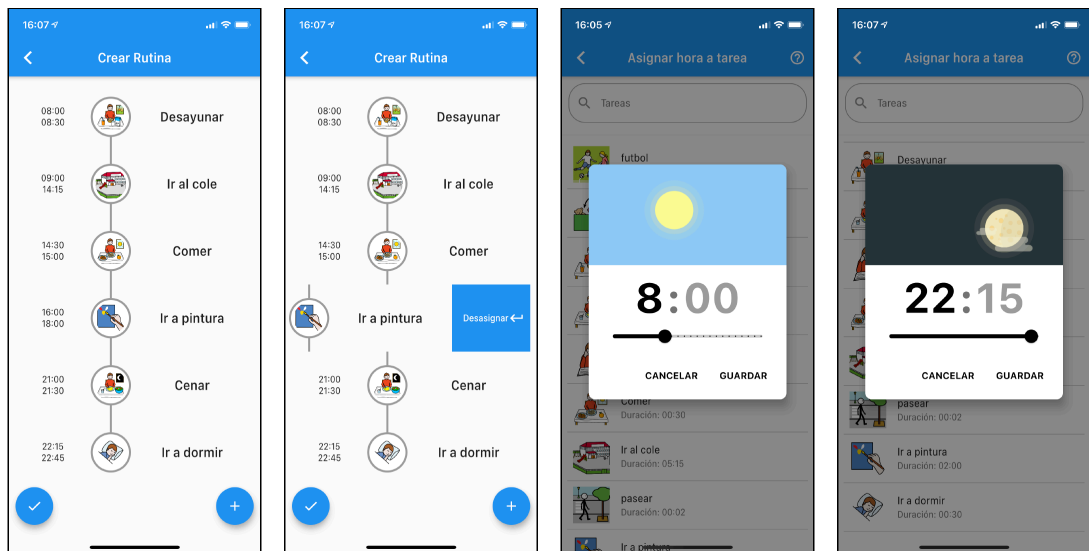


Figura 9.3: Capturas da listaxe de tarefas dentro da rutina correndo na aplicación real.

Neste punto da aplicación fíxose unha decisión de implementación sobre o formato no cal que se gardaría a duración das tarefas e o formato dos momentos (horas) en que cada tarefa está asignada dentro dunha rutina.

9.5.2 Formato do campo duración das tarefas

Tras varias probas sobre o formato do campo duración das tarefas, decidiuse que o mellor formato para o almacenamento na base de datos do campo duración da tarefa era un String coa forma 'HH:mm'. En primeiro lugar pensárase nun formato tipo DateTime pero, debido a que na aplicación se mostra varias veces o campo duración da tarefa, decidiuse almacenalo co tipo String e coa forma mencionada anteriormente para non ter que cambiar o formato e recortar o dato obtido de tipo DateTime cada vez que fose necesario mostrar ou operar coa duración.

9.5.3 Formato dos momentos (fin e comezo) de asignación das tarefas dentro dunha rutina

Entender a decisión de implementación do formato do campo duración das tarefas é necesario para comprender tamén a decisión de implementación dos momentos de fin e de comezo das tarefas dentro das rutinas.

Como se explicará na sección 9.5.4, é necesario comparar varias veces as horas das tarefas entre elas dentro das rutinas á hora de crealas ou de editalas. Como Dart non contén un formato básico para tempo (unicamente contén un formato DateTime que tamén contén unha data), a decisión de implementación para gardar esta información consistiu en asignar un día xenérico para todas as tarefas cando son creadas ou editadas. Esta decisión unicamente se tomou para poder facer unha comparación entre tarefas máis doada xa que, simplemente, se necesitan comparar horas para detectar se as tarefas se solapan en tempo dentro dunha determinada rutina.

Desta maneira, o día escollido (sen maior importancia) foi o día da aterraxe do home na lúa, ao cal se engade a hora seleccionada de inicio da tarefa, obtendo así o campo *inicio* do tipo DateTime que representa o instante no que se inicia a tarefa. A este campo, pode agora engadirselle as horas ('HH') e os minutos ('mm') do campo *duracion* da tarefa para obter así tamén o campo *fin* de tipo DateTime que representa o instante no que finaliza a tarefa.

9.5.4 Compatibilidade das tarefas dentro da rutina

Cando se engaden tarefas ás rutinas, hai que comprobar que as tarefas non se superpoñen, debido a que non pode haber dúas tarefas nunha mesma franxa horaria.

A maneira na que a aplicación xestiona isto é obtendo unha listaxe ordenada pola hora de inicio das tarefas xa asignadas á rutina (proporcionada por Firebase) e comprobando se o intervalo desde a hora de inicio da nova tarefa ata a súa hora de fin coincide con algún dos intervalos das tarefas xa asignadas. No caso de que coincida, a aplicación mostrará unha mensaxe de erro, non deixando inserir a tarefa a esa hora porque a franxa horaria especificada

xa estaría ocupada por outra tarefa.

9.6 Implementación das estatísticas

A implementación das estatísticas, como xa se anticipou no capítulo de deseño, tamén variou con respecto ao deseño inicial xa que, no canto de amosar as mesmas estatísticas de diferentes xeitos, como podería ser mediante un gráfico de barras, un gráfico de área ou gráfico de queixiños, as estatísticas visualízanse unicamente mediante un gráfico de barras pero este gráfico pasa a ser interactivo, permitindo unha selección por parte do usuario administrador.

Desta maneira, ademais de poder mostrar unicamente a estatística ou subconxunto de estatísticas desexadas (e non todas ao mesmo tempo), tamén se pode elixir máis dunha tarefa para comparar entre si diferentes estatísticas para diferentes tarefas.

En concreto, as estatísticas que permite visualizar a aplicación son: a porcentaxe do tempo empregada para completar as tarefas respecto do programado, a porcentaxe das tarefas ben feitas e a porcentaxe das tarefas que non se completaron segundo o previsto.

A implementación do gráfico interactivo realizouse coa librería oficial proporcionada por Flutter `charts_flutter`, que permite a definición de diferentes gráficos. A maneira de implementar este gráfico coas estatísticas foi a seguinte:

- Ao seleccionar unha tarefa realízanse as operacións necesarias para obter as súas estatísticas correctamente.
- Créase unha instancia dunha clase `TaskStatistics` cuns atributos que son o nome da tarefa e as diferentes estatísticas.
- Engádese esta instancia a unha listaxe de `TaskStatistics` que será a utilizada polo gráfico.
- Defínese o tipo de gráfico, así como o número de entradas que vai ter (lonxitude da listaxe de `TaskStatistics`) e que datos corresponden a cada cousa dentro do gráfico (atributos das diferentes instancias `TaskStatistics`).
- Impleméntase o xeito no que se vai a comportar o gráfico (neste caso permitindo mostrar ou ocultar diferentes tipos de datos).
- Realízase un `SetState` que permite que se recargue a pantalla actual co gráfico con novos datos.

9.6.1 Tempo utilizado

Esta estatística está baseada no tempo utilizado para completar cada tarefa. No momento en que o usuario indica que finalizou a tarefa gárdase un rexistro na base de datos da porcentaxe de tempo utilizado desde o inicio da tarefa ata ese momento con respecto á duración

total da tarefa. Grazas á maneira na que se almacena esta porcentaxe na base de datos, logo pódense obter todas as porcentaxes que se foron gardando para unha determinada tarefa para despois facer a media entre elas. Esa media será a porcentaxe final mostrada nesta estatística.

9.6.2 Tarefas ben feitas

Coma se explicará máis detalladamente no capítulo de probas, algunhas tarefas poden ser avaliadas ao finalizar. Esta estatística é a porcentaxe do número de veces que unha tarefa se realizou correctamente sobre o número de veces totais que se realizou esa tarefa.

9.6.3 Tarefas mal feitas

De maneira contraria á estatística de tarefas ben feitas, esta estatística é outra porcentaxe que mostra a porcentaxe do número de veces que unha tarefa non se realizou correctamente sobre o número total de veces que se realizou a tarefa.

Na figura 9.4, pode observarse o resultado visual, a forma na que se presentan as estatísticas e o xeito de interaccionar coas diferentes gráficas.

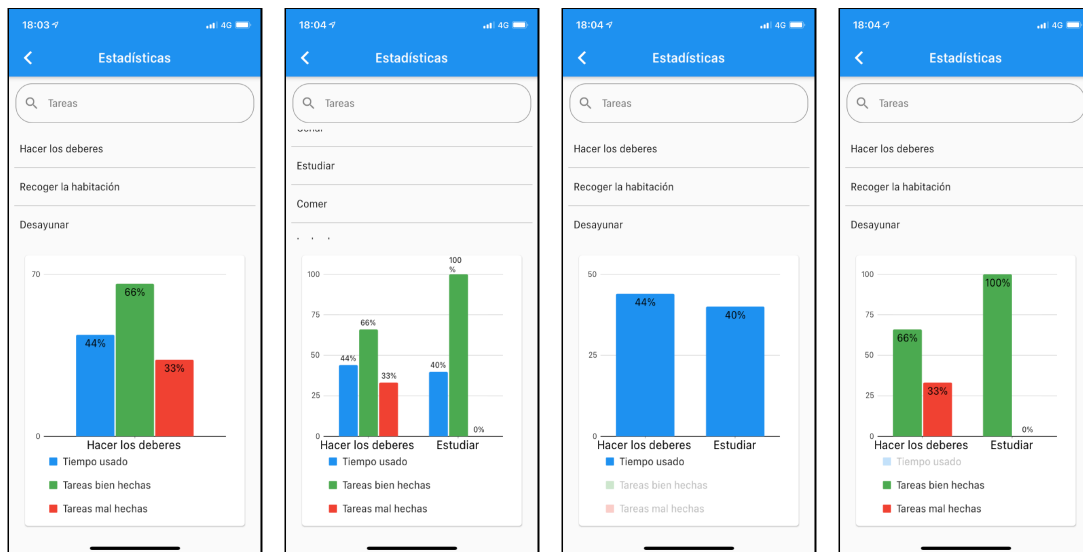


Figura 9.4: Capturas das estatísticas correndo na aplicación real.

9.6.4 Estatísticas dentro da base de datos

Para incluír as estatísticas dentro da base de datos, o xeito implementado foi mediante a utilización, dependendo do tipo de estatística (estatística de tempo utilizado ou estatística de tarefa ben ou mal feita) dunha subcolección ou doutra dentro do campo das estatísticas.

Subcoleccións por cada tarefa

Como unha mesma tarefa pode ser realizada máis dunha vez durante as rutinas, o xeito de almacenar os datos estatísticos na base de datos é gardando un documento cun identificador único por cada vez que se realiza unha tarefa dentro doutra subcolección, co seu identificador de tarefa como chave. Desta maneira dentro da subcolección correspondente ás porcentaxes estarán todos os identificadores das tarefas realizadas, e dentro de cada identificador das tarefas haberá tantos documentos como número de veces fora realizada esa tarefa. A implementación segue o mesmo esquema para as estatísticas correspondentes á súa avaliación.

9.7 Programación asíncrona

Debido ao uso dun modelo aloxado na nube e á utilización dunha API externa, a aplicación necesita utilizar continuamente funcións de programación asíncronas. Deste xeito, este tipo de programación ten unha grande importancia na aplicación.

A programación asíncrona caracterízase pola espera a que se complete unha operación, normalmente de lectura/escritura ou de conexión de rede. Desta maneira, pódese bloquear a execución ata que se complete a tarefa.

Todas as conexións realizadas dende o controlador con Firebase e cos seus módulos foron implementados seguindo este tipo de programación. Tamén foi empregada para realizar as peticións á API de ARASAAC.

Un exemplo do uso de programación asíncrona no código pode ser a maneira de pedir á base de datos a URL dun pictograma que, ademais, se ten que utilizar en varias ocasións. A función para realizar esta tarefa definiríase así:

```
1 Future<String> getTaskPicto(firestoreInstance, auth, taskId) async{
2   try {
3     FirebaseUser _user = await auth.getCurrentUser();
4     String _userId = _user.uid;
5     var data = await firestoreInstance
6       .collection('administradores')
7       .document(_userId)
8       .get();
9     String _kidId = data.data['usuario'];
10    var task = await firestoreInstance
11      .collection("usuarios")
12      .document(_kidId)
13      .collection('tarefas')
14      .document(taskId).get();
15    return (task["pictograma"]);
16  } catch (e) {
```

```
17     print('Error: $e');  
18   }  
19 }
```

E chamaríase da seguinte maneira:

```
1 String picto = await getTaskPicto(firestoreInstance, auth, id);
```

9.7.1 Refresco da pantalla do neno

A maneira de interaccionar o neno coa pantalla principal é, por un lado, mediante a visualización da tarefa actual ou da súa correspondente barra de progreso e, por outro lado, mediante as diferentes accións que pode realizar sobre as tarefas: mostrar as restantes, mostrar a descrición ou indicar que xa rematou a actual.

Debido a estas posibilidades de interacción e a que as tarefas están definidas por minutos e non por segundos (i.e., unha tarefa non pode durar 20 minutos **e 30 segundos** ou comezar ás 10:00:**20** por exemplo), implementouse unha acción de refresco desta pantalla cada vez que cambian os minutos ou cada vez que o neno interacciona coa aplicación. Aínda que unha tarefa pode durar bastante tempo e, polo tanto, pode pasar un certo tempo sen que o neno interactúe directamente coa aplicación, é necesario esta acción de refresco para actualizar a barra de progreso que lle indica visualmente ao neno cando queda para rematar.

Esta decisión de implementación tomouse para que garantir que a aplicación se executa de forma fluída tamén en dispositivos con poucos recursos.

NESTE capítulo explicaranse as diferentes probas realizadas na aplicación desenvolvida neste TFG e como se mellorou a calidade da mesma logo destas probas.

10.1 Probas funcionais

As probas funcionais son as probas baseadas na execución, revisión e retroalimentación das funcionalidades previamente deseñadas para o software. Existen probas funcionais de diferentes tipos. No caso deste TFG, as probas funcionais que se realizaron foron as seguintes.

10.1.1 Probas unitarias

As probas unitarias son aquelas encargadas de comprobar o correcto funcionamento dunha unidade de código. No caso da aplicación deste TFG, as unidades de código son as diferentes funcións e os diferentes *widgets* implementados.

Dentro das diferentes probas unitarias que se fixeron destacan dúas polas modificacións que conlevaron estas probas:

Probas unitarias para a listaxe de tarefas e para a listaxe de rutinas

Tanto no banco de tarefas coma no banco de rutinas lístanse, respectivamente, as tarefas e as rutinas dispoñibles. A primeira implementación desta listaxe realizaba a petición á base de datos cunha función de tipo *future*. Desta maneira, o comportamento das funcións encargadas da listaxe era que, á hora de cargar as tarefas ou as rutinas, facíase unha petición á base de datos para obter os datos das mesmas. O problema disto foi que, ao existir a posibilidade de que dous administradores estean engadindo ou eliminando ao mesmo tempo tarefas ou rutinas, estes non poderían ver reflectida na listaxe a acción que acababa de realizar o outro ata que se volvese facer outra petición á base de datos. Este problema foi descuberto nesta fase de probas e o xeito de solucionalo foi a modificación do modo de obter a listaxe da base de

datos: no canto de realizar a petición cun *future*, a petición realizouse cun *stream*. Un *stream* funciona de xeito moi similar a un *future* e a única diferenza é que, cando hai un cambio na base de datos, a propia base de datos avisa á listaxe de que acaba de modificarse, polo que a listaxe volve facer unha petición automaticamente.

10.1.2 Probas de integración

As probas de integración son aquelas que, unha vez finalizadas as probas unitarias, se encargan de comprobar o correcto funcionamento das diferentes unidades de código unha vez son integradas conxuntamente.

No caso deste TFG, todas as probas de integración que se realizaron tiveron un resultado positivo, polo que non hai ningunha proba que mereza ser destacada.

10.1.3 Probas de aceptación

As probas de aceptación son as que determinan se o software desenvolvido cumpre coas necesidades ou requisitos dos usuarios.

Dentro deste TFG, as probas de aceptación realizáronse nos dous últimos *sprints* do proxecto, momentos do desenvolvemento nos que a aplicación xa era suficientemente funcional para realizar este tipo de probas.

Para estas probas contouse co apoio e axuda dunha entidade da nosa contorna para obter un *feedback* real. Esta entidade é unha asociación especializada no ámbito da atención á diversidade e, en especial, nos trastornos da linguaxe e comunicación, nos trastornos xerais do desenvolvemento e no autismo. Esta asociación dispón e ofrece a colaboración de varios profesionais que levan os programas de atención e intervención a domicilio (psicolóxica, psicopedagóxica, logopédica, psicomotricidade...), programas de musicoterapia, de ocio e tempo de lecer, entre outros, e nos que participan persoas con todo tipo de diversidade funcionais.

Primeira proba de aceptación

A primeira proba de aceptación realizouse entregando unha versión da aplicación aos titores deste TFG, obtendo un *feedback* dos mesmos e outro por parte da asociación grazas á segunda reunión, explicada en detalle no anexo B, realizada entre os titores e os profesionais da mesma. A realimentación recibida logo desta primeira proba derivou na integración dos seguintes cambios na aplicación:

1. Permitir ao neno cambiar a cor da aplicación.
2. Engadir unha barra de progreso en lugar do reloxo para a pantalla inicial.
3. Non permitir rematar instantaneamente as tarefas.

4. Posibilidade de avaliar as tarefas.

Estes cambios foron incluídos con éxito na aplicación.

Segunda proba de aceptación

A segunda proba de aceptación realizouse facendo unha demo e entregando unha versión da aplicación a dúas educadoras da entidade, obtendo así un *feedback* directo.

A demo e as conclusións desta reunión están explicadas en maior detalle no apéndice C. Este *feedback* deu lugar a posibles solucións que aínda non se implantaron na aplicación quedando como traballo futuro do que se falará no capítulo 11 de Conclusións.

Nesta reunión tamén se chegou ao acordo de que se vai probar a aplicación tanto con nenos no entorno familiar grazas á colaboración das familias que traballan coa asociación como nas terapias que van levar a cabo durante as primeiras semanas de setembro deste ano. É importante a proba durante un período un pouco longo de tempo porque nos TEA non se pode probar puntualmente a aplicación xa que non ofrecería ningún tipo de información.

10.2 Probas non funcionais

As probas non funcionais son as probas utilizadas para verificar o correcto funcionamento e cumprimento dos requisitos non funcionais. Realizáronse as seguintes probas para este tipo de requisitos:

10.2.1 Probas de compatibilidade

Realizáronse probas da aplicación en diferentes dispositivos Android e iOS con diferentes versións do sistema operativo en cada caso.

Versións do sistema operativo

Probáronse diferentes versións dos dous sistemas operativos compatibles (Android e iOS). Entre estas probas destacan a proba nun dispositivo con iOS 11.4 para a confirmación de que a aplicación non funciona en versións inferiores, e a proba nun dispositivo con Android 7.0, confirmando o mesmo no sistema operativo Android.

Por outro lado, probouse tamén nun dispositivo iOS físico cunha versión beta de iOS 14 (actualmente a última versión estable de iOS é a 13.4) con éxito, amosando así a compatibilidade futura da aplicación.

Finalmente, coas últimas versións estables de iOS (13.4) e Android (10) a aplicación funciona perfectamente.

10.2.2 Probas de usabilidade

Realizáronse diferentes probas de usabilidade con dispositivos de diferente tamaño, resolución e cociente de pantalla.

Dimensións

A aplicación probouse en dispositivos de diferentes dimensións e resolucións, desde dispositivos pequenos de 4 polgadas ata tabletas de 10 polgadas.

Dentro destas probas, observouse un funcionamento ideal en cocientes de pantalla 16:9 (cociente de case todos os teléfonos móbiles) e un funcionamento incorrecto da aplicación en cocientes de pantalla 4:3, como poden ser os dalgunhas tabletas, debido a que era preciso facer uso do *scroll* para poder mostrar todos os compoñentes visuais da aplicación. Esta circunstancia fixo cambiar a maneira na que os *widgets* están implementados, redimensionándoos dependendo da resolución e do formato que ten o dispositivo no que está a correr a aplicación.

Conclusiones

NESTE capítulo, inclúense as principais conclusións e reflexións persoais logo da finalización do TFG, así como as principais liñas futuras de traballo para a mellora da aplicación.

11.1 Conclusións

Actualmente dispónse dunha aplicación funcional resultado de todo o traballo realizado neste TFG, polo que os obxectivos expostos ao principio da memoria foron conseguidos.

Cubríronse as competencias da titulación en xeral, aplicando metodoloxías, ferramentas e tecnoloxías aprendidas durante o transcurso da carreira para a realización e obtención exitosa desta aplicación. Así, a realización deste TFG posibilitoume a aprendizaxe do uso real de metodoloxías áxiles e da planificación dun proxecto software, e permitiu realizar un seguimento facendo fronte a factores externos que afectan no desenvolvemento dun proxecto real. E, finalmente, posibilitou a aplicación das competencias adquiridas durante a carreira a un tema parcialmente descoñecido como é a atención á diversidade no caso de nenos con TEA mediante unha aplicación móbil. Isto esixiu unha forte documentación e busca de información previa, para así poder ofrecer unha solución máis adaptada ás necesidades deste colectivo mediante unha aplicación que responda do mellor xeito posible ás características específicas destes nenos.

Como experiencia persoal, o tempo invertido neste traballo servíume para especializarme en certas tecnoloxías nas que unicamente tiña uns coñecementos iniciais básicos, como era o desenvolvemento para aplicacións móbiles. A maior dificultade dentro do proxecto foi o deseño de solucións para certos problemas concretos expostos que, aínda que unha vez deseñadas e implementadas parezan doadas, non o foron. Un exemplo destas solucións especialmente complicadas de deseñar foi a maneira de xestionar as rutinas, o que ocorre dentro delas cando non hai tarefas ou como evitar superpoñer dentro das rutinas as tarefas de diferente duración e diferente comezo e fin.

Nos próximos meses procurarase que esta aplicación siga medrando, engadindo novas funcionalidades e pulindo as actuais para servir e ocupar un espazo dentro da educación e vida cotiá dos nenos con TEA que ningunha outra ferramenta cubre do mesmo xeito nestes momentos. Neste sentido, a excelente acollida da aplicación logo das probas realizadas cunha entidade da nosa contorno, tal e como se indicaba no capítulo anterior, demostran a necesidade da mesma para a atención á diversidade dos nenos con TEA tanto no ámbito escolar como nestas asociacións.

Unha conclusión obtida logo do traballo nestes últimos meses, debido á situación que estamos a vivir por mor da Covid-19, é que aplicacións coma esta son moi útiles para que os educadores que traballan con estes nenos, e que ao mellor non poden facelo de xeito presencial ou que si poden facelo de xeito presencial pero cunha asiduidade menor á dunha situación normal, poidan seguir axudando e monitorando a estes nenos.

11.2 Comparativa

Logo do desenvolvemento do proxecto, decidiuse engadir á táboa 3.1 do capítulo 3, correspondente ao Estado da Arte, unha fila coa aplicación desenvolvida neste TFG para mostrar as características que esta ofrece e comparalas coas das alternativas existentes.

Característica Aplicación	Inclusión de Pictogramas	Definición de Rutinas	Seguimento de Tarefas	Consulta de Estatísticas	Sistema Operativo	Prezo
Día a Día	✓	✓	✗	✗	iOS e Android	Gratuíta
In-Tic Agenda	✓	✓	✗	✗	Windows	Gratuíta
Diario de Autismo	✗	✗	✗	✓	Android	Gratuíta
ChoiceWorks	✓	✓	✓	✗	iOS	9.99\$
Pictorario	✓	✓	✗	✗	Android	Gratuíta
Aplicación deste TFG	✓	✓	✓	✓	iOS e Android	Gratuíta

Táboa 11.1: Táboa comparativa das características das aplicacións existentes.

Como pode verse nesta táboa, a aplicación desenvolvida recolle as vantaxes de todas as aplicacións existentes e incorpora todas as súas funcionalidades básicas.

11.3 Liñas futuras

Como se comentou anteriormente, a idea é seguir mellorando a aplicación no futuro e incorporando novas funcionalidades que poidan resultar de utilidade. Estas melloras determinan as liñas de traballo futuras sobre a base do traballo realizado neste TFG.

En primeiro lugar, convén recordar que a aplicación está implementada unicamente en castelán. Polo tanto, unha primeira liña sería a internacionalización da aplicación para que

sexa posible seleccionar o idioma dentro da configuración da propia aplicación, facendo sentir así máis cómodo ao usuario final co uso da mesma.

A seguinte liña futura a abarcar sería engadir as propostas que xurdiron na reunión exposta no apéndice C, que tivo lugar coas educadoras da entidade no mes de agosto. Estas funcionalidades son basicamente as seguintes:

- Engadir sons personalizables para o remate exitoso dunha tarefa.
- Poder editar a tarefa xenérica que se mostra cando non hai ningunha tarefa no momento actual.
- Incluír unha cualificación numérica para a avaliación dunha tarefa. Deste xeito teríamos unha cualificación cuantitativa no canto da cualitativa implementada ata agora, onde unicamente se indica se se fixo ben ou mal a tarefa.

Outra liña futura sería a inclusión de novas opcións para as estatísticas da aplicación, permitindo a súa exportación a un ficheiro externo para a posterior manipulación ou a obtención doutro tipo de gráficas máis específicas, como podería ser unha liña que reflecta a evolución nun período que poida seleccionar o usuario administrador.

Por último, como liña futura tamén se inclúe a publicación nas tendas de aplicacións para os sistemas operativos móbiles. Debido ao custo que ten a publicación nas diferentes tendas (Play Store para Androd e App Store para iOS), a aplicación aínda non está publicada comercialmente en ningunha das dúas tendas de aplicacións.

Apéndices

Acta de reunión inicial

Resumen acta de reunión inicial:

Fecha: 22 de julio de 2020

Asistentes:

1. Óscar Fresnedo Arias
 2. Psicóloga da asociación colaboradora
-

Reunión por videoconferencia en la que se abordan las siguientes cuestiones:

- Se le presenta a la persona de contacto de la asociación la idea y el alcance del proyecto a realizar.
- Se define en qué va a consistir la colaboración y se programan una serie de reuniones en las que se llevarán a cabo las actividades acordadas para articular la colaboración. Básicamente, se programan dos reuniones: en la primera se realizará un seguimiento del proyecto, donde los profesionales probarán una primera versión funcional de la aplicación y aportarán su feedback para refinar y mejorar lo que consideren oportuno; y en la segunda reunión, los profesionales probarán junto con algunos niños la versión definitiva de la aplicación para hacer una valoración de su utilidad y usabilidad en un contexto práctico.
- Se le explican a la persona de contacto las principales funcionalidades de la aplicación. Además, se consultan con ella ciertos detalles relativos a la implementación concreta de algunas funcionalidades, especialmente aquellas que tienen que ver con la presentación de la información a los niños: progreso de las tareas, notificación de finalización de tareas, información en la descripción de las tareas...

Acta de reunión de seguimiento

Resumen acta de reunión de seguimiento:

Fecha: 24 de agosto de 2020

Asistentes:

1. Óscar Fresnedo Arias
 2. Psicóloga da asociación colaboradora
 3. Logopeda da asociación colaboradora
-

Reunión presencial de seguimiento en la que se realizaron las siguientes actividades:

- Se presenta a las dos profesionales de la asociación una primera versión funcional de la misma desarrollada en el proyecto.
- Se hace una demostración de las principales funcionalidades de la aplicación, tanto de la parte de administración, más orientada a los cuidadores/padres, como de la parte orientada a la visualización de tareas por los niños con TEA.
- Las profesionales valoran en general la aplicación y, en particular, las distintas funcionalidades, aportando un feedback muy valioso para corregir y mejorar ciertos detalles que puedan ayudar a construir una aplicación más usable y que resulte más útil tanto a cuidadores como a niños.

Las principales conclusiones alcanzadas tras la reunión se pueden resumir en los siguientes puntos:

-
- La valoración global de la aplicación de los profesionales ha sido muy positiva, destacando que puede ayudar a mejorar, por un lado, el grado de autonomía de los niños con TEA y, por otro, proporcionar información muy interesante a los cuidadores/profesionales sobre cómo los niños han completado las diferentes rutinas / tareas. Esta información puede ser luego empleada para ajustar las diferentes tareas en las rutinas para los niños.
 - Además, han destacado lo adecuadas que resultan este tipo de aplicaciones para niños con estas características, ya que representan un estímulo visual mucho más atractivo que las tradicionales agendas en papel. Por último, han destacado la flexibilidad que proporciona a la hora de definir rutinas por la disponibilidad de añadir pictogramas de forma instantánea, sin necesidad de crear la agenda y los pictogramas en formato físico.
 - Se han recogido además una serie de sugerencias proporcionadas por los profesionales de la asociación:
 - Añadir la posibilidad de incorporar audio a la descripción de las tareas para adaptarse a las distintas capacidades/sensibilidades de los niños con TEA.
 - Bloquear un tiempo el botón de finalización de tarea para que los niños no puedan terminar una tarea que no quieren realizar al principio, o para evitar finalizar tarea en caso de que le den accidentalmente al presentarse el pictograma con la tarea.
 - Incorporar una tarea de recompensa configurable si acaban la tarea antes del tiempo predeterminado.
 - Incorporar un botón/pop-up para que el cuidador pueda evaluar si el niño ha realizado correctamente una tarea (de trabajo). Esta información será recogida por la aplicación y se visualizará en el apartado de estadísticas.
 - Incluir una barra/icono de progreso para que el niño pueda anticipar el final de la tarea o conocer cuanto tiempo resta. Este icono será descriptivo para evitar distracciones.
 - Activar una alarma con un sonido agradable para indicarle al niño cuando finaliza una tarea.
 - Cambiar la tipografía en las pantallas de visualización para el niño. Usar preferiblemente mayúsculas.
 - Se han definido qué estadísticas sería interesante que mostrara la aplicación para los cuidadores/profesionales.

Acta de reunión final

Resumen acta de reunión final:

Fecha: 01 de septiembre de 2020

Asistentes:

1. Noé Vila Muñoz
 2. Óscar Fresnedo Arias
 3. Psicóloga da asociación colaboradora
 4. Logopeda da asociación colaboradora
 5. Nai dun neno con TEA
 6. Neno con TEA
-

Reunión presencial en la que se realizaron las siguientes actividades:

- Se presenta a las dos profesionales de la asociación DISMACOR la aplicación desarrollada con todas las funcionalidades.
- El alumno explica en detalle todas las funcionalidades de la aplicación, centrándose en aquellas que se han incorporado desde la última reunión.
- Las profesionales comentan las dudas sobre el uso e instalación de la aplicación con el alumno y su tutor.

-
- Las profesionales prueban las diferentes funcionalidades de la aplicación sobre un iPad con el asesoramiento del alumno.
 - Al final de la reunión, se le muestra la aplicación a un niño con TEA y, bajo la supervisión de su madre y las profesionales, el niño interactúa durante un periodo de tiempo breve con la aplicación.

Una vez completada la reunión se establecen las siguientes conclusiones:

- De nuevo, las profesionales han valorado de forma positiva la aplicación y han expresado que puede resultar útil tanto en el ámbito más familiar como a la hora de realizar intervenciones específicas sobre niños con TEA.
- De igual modo, han mostrado su conformidad con los cambios incorporados en la aplicación siguiendo sus recomendaciones en la reunión de seguimiento.
- Además, las profesionales han propuesto una serie de sugerencias relacionadas con las nuevas funcionalidades añadidas en la aplicación desde la última reunión y han matizado algunos detalles sobre las recomendaciones de la reunión anterior. Entre estas sugerencias, se puede destacar:
 - Posibilidad de valorar las tareas realizadas por el niño de acuerdo a una escala numérica, en lugar de establecer simplemente que la tarea se ha hecho perfecta o se ha realizado de forma incorrecta.
 - Incorporar esta información en la parte de estadísticas para que el profesional pueda consultar estos datos de forma automática.
 - Añadir un cuadro de observaciones en la vista para la valoración de tareas, donde el educador o terapeuta pueda añadir detalles que ha observado durante la realización de la tarea.
 - Posibilidad de personalizar el sonido que el niño escucha cuando termina el tiempo programado para una tarea.
- Por último, se ha definido en detalle cómo se realizarán las pruebas de la aplicación con un pequeño grupo de niños con TEA.

En la reunión inicial, ya se estableció que no resultaba demasiado útil que los niños probarán la aplicación en una única sesión y de forma puntual debido al alcance del proyecto y a las características de los niños con TEA. Este hecho se ha comprobado de forma práctica en la reunión cuando se le ha dejado interactuar a un niño con la aplicación. El objetivo de esta prueba era simplemente comprobar si la interfaz resultaba agradable e intuitiva, pero se ha comprobado que un niño afectado con TEA necesita

un cierto tiempo para familiarizarse con la aplicación, además de estar en un entorno controlado y tranquilo.

Teniendo en cuenta esto y siguiendo la idea inicial, se ha decidido enviar la aplicación a las familias de los niños que van a probarla para que, a lo largo de las dos próximas semanas, prueban y valoren el uso de la herramienta en un entorno familiar. Además, las profesionales se han comprometido a utilizar la aplicación también en sus intervenciones durante las dos próximas semanas con esos niños para valorar su utilidad en un entorno educativo y desde un punto de vista profesional.

Al final de este período de prueba, se utilizarán unas encuestas diseñadas para que, tanto las familias como los profesionales, nos den su opinión sobre diferentes aspectos de la aplicación y su uso por parte de los niños con TEA en los dos tipos de entornos para los que está pensada.

Relación de Acrónimos

OMS *Organización Mundial da Saúde.*

TEA *Trastorno do Espectro Autista.*

INE *Instituto Nacional de Estadística.*

CIE-11 *Clasificación Internacional de Enfermedades, 11.a revisión.*

ARASAAC *Aragón. Sistemas Aumentativos y Alternativos de Comunicación.*

IDE *Integrated Development Environment.*

GPS *Global Positioning System.*

MVC *Modelo Vista Controlador.*

URL *Uniform Resource Locator.*

HTTP *Hypertext Transfer Protocol.*

API *Application Programming Interfaces.*

PNG *Portable Network Graphics.*

Bibliografía

- [1] Nhungdo, “Khung thời gian cho một sprint.jpg,” 2016. [Online]. Available: https://commons.wikimedia.org/wiki/File:Khung_thời_gian_cho_một_Sprint.jpg
- [2] D. Zager, *Autism Spectrum Disorders: Identification, Education and Treatment*, 3rd ed. Lawrence Erlbaum Associates Publishers, 2005.
- [3] J. Baker, *The Autism Social Skills Picture Book: Teaching Communication, Play and Emotion*. Future Horizons Inc., 2001.
- [4] I. N. de Estadística, “Encuesta sobre equipamiento y uso de tecnologías de información y comunicación en los hogares 2016,” 2016. [En línea]. Disponible en: https://www.ine.es/jaxi/Tabla.htm?path=/t25/p450/base_2011/a2016/l0/&file=08029.px&L=0
- [5] “Autism society of america,” 2020. [En línea]. Disponible en: <https://www.autism-society.org>
- [6] I. N. de Estadística, “Tasa de población con discapacidad que tiene diagnosticadas determinadas enfermedades crónicas según la enfermedad por ccaa y sexo,” 2008. [En línea]. Disponible en: <https://www.ine.es/jaxi/Datos.htm?path=/t15/p418/a2008/hogares/p02/modulo1/l0/&file=04028.px#!tabs-tabla>
- [7] —, “Módulo de discapacidades y deficiencias 1999,” 1999. [En línea]. Disponible en: <https://www.ine.es/dynt3/inebase/es/index.htm?type=pcaxis&path=/t15/p418/a1999&file=pcaxis>
- [8] “Confederación autismo españa,” 2020. [En línea]. Disponible en: <https://www.autismo.org.es>
- [9] “Asociación de padres de personas con autismo,” 2020. [En línea]. Disponible en: <https://www.apna.es>
- [10] “Federación española de autismo,” 2020. [En línea]. Disponible en: <https://www.fespau.es>

- [11] “World autism organisation,” 2020. [En línea]. Disponible en: <https://worldautismorganisation.com>
- [12] “Autism speaks,” 2020. [En línea]. Disponible en: <https://www.autismspeaks.org>
- [13] S. Panerai, M. Zingale, G. Trubia, M. Finocchiaro, R. Zuccarello, R. Ferri, and M. Elia, “Special education versus inclusive education: The role of the teacch program,” *Journal of Autism and Developmental Disorders*, vol. 39, p. 9, 2009.
- [14] L. V. V., “Autism as a model of abnormal emotional development,” *Psychology in Russia: State of the art*, vol. 2, 2009.
- [15] J. Niemczyk, R. Fischer, C. Wagner, A. Burau, T. Link, and A. von Gontard, “Detailed assessment of incontinence, psychological problems and parental stress in children with autism spectrum disorder,” *Journal of Autism and Developmental Disorders*, vol. 49, 2009.
- [16] “Leo kanner,” 2020. [En línea]. Disponible en: https://en.wikipedia.org/wiki/Leo_Kanner
- [17] “Hans asperger,” 2020. [En línea]. Disponible en: https://en.wikipedia.org/wiki/Hans_Aasperger
- [18] L. Kanner, “Autistic disturbances of affective contact.” [En línea]. Disponible en: http://mail.neurodiversity.com/library_kanner_1943.pdf
- [19] J. L. Skues and E. G. Cunningham, “A contemporary review of the definition, prevalence, identification and support of learning disabilities in australian schools,” *Australian Journal of Learning Difficulties*, vol. 16, no. 2, pp. 159–180, 2011. [En línea]. Disponible en: <https://doi.org/10.1080/19404158.2011.605154>
- [20] J. Reilly, *Language Disorders*, 06 2016.
- [21] “Cie-11,” 2019. [En línea]. Disponible en: <https://icd.who.int/es>
- [22] “Autism society of america, asperger syndrome,” 2020. [En línea]. Disponible en: <https://www.autism-society.org/what-is/aspergers-syndrome/>
- [23] D. M. Lowth, “Childhood disintegrative disorder(heller’s syndrome,” 2014. [En línea]. Disponible en: <https://patient.info/doctor/childhood-disintegrative-disorder-hellers-syndrome>
- [24] F. Orange, “Día a día.” [En línea]. Disponible en: <http://www.fundacionorange.es/aplicaciones/dia-a-dia/>

- [25] —, “In-tic agenda.” [En línea]. Disponible en: <http://www.fundacionorange.es/aplicaciones/in-tic-agenda/>
- [26] A. Mekla, “Diario de autismo.” [En línea]. Disponible en: <https://play.google.com/store/apps/details?id=com.wellbeingdiaries.autismdiary&hl=es>
- [27] B. V. LLC, “Choiceworks.” [En línea]. Disponible en: <https://apps.apple.com/us/app/choiceworks/id486210964>
- [28] J. Prieto, “Pictorario.” [En línea]. Disponible en: <https://play.google.com/store/apps/details?id=javi.prieto.pictorario>
- [29] S. Palao, “Pictogramas arasaac.” [En línea]. Disponible en: <http://arasaac.org>
- [30] IntelliJ idea ultimate edition. [En línea]. Disponible en: <https://www.jetbrains.com/es-es/idea/>
- [31] Android studio. [En línea]. Disponible en: <https://www.developer.android.com/studio>
- [32] Xcode. [En línea]. Disponible en: <https://developer.apple.com/xcode/>
- [33] Dart. [En línea]. Disponible en: <https://dart.dev>
- [34] Flutter. [En línea]. Disponible en: <https://flutter.dev>
- [35] Firebase. [En línea]. Disponible en: <https://firebase.google.com>
- [36] Cloud firestore. [En línea]. Disponible en: <https://firebase.google.com/docs/firestore>
- [37] Google cloud storage. [En línea]. Disponible en: <https://cloud.google.com/storage>
- [38] Firebase authentication. [En línea]. Disponible en: <https://firebase.google.com/docs/auth>
- [39] Android. [En línea]. Disponible en: <https://www.android.com>
- [40] iOS. [En línea]. Disponible en: <https://www.apple.com/es/ios/ios-13>
- [41] The latex project. [En línea]. Disponible en: <https://www.latex-project.org>
- [42] Git. [En línea]. Disponible en: <https://git-scm.com>
- [43] Microsoft project. [En línea]. Disponible en: <https://products.office.com/en-us/project/project-management-software>
- [44] Visual paradigm online diagrams. [En línea]. Disponible en: <https://online.visual-paradigm.com>

- [45] Balsamiq mockups. [En línea]. Disponible en: <https://balsamiq.com>
- [46] J. Sutherland, *SCRUM, the Art of Doing Twice the Job in Half the Time*, 1st ed. Crown Business, 2014.
- [47] Salario medio scrum master. [En línea]. Disponible en: <https://www.indeed.es/salaries/scrum-master-Salaries>
- [48] Salario medio analista. [En línea]. Disponible en: <https://www.indeed.es/salaries/analista-de-sistema-Salaries>
- [49] Salario medio diseñador. [En línea]. Disponible en: <https://www.indeed.es/salaries/diseñador-web-Salaries>
- [50] Salario medio desarrollador. [En línea]. Disponible en: <https://www.indeed.es/salaries/desarrollador-de-software-Salaries>
- [51] U. Frith, *Autism: A Very Short Introduction*. Oxford, 2005.
- [52] C. A. España, “Estudio sociodemográfico sobre las personas con tea,” 2019. [En línea]. Disponible en: <http://www.autismo.org.es/actualidad/articulo/el-estudio-sociodemografico-desarrollado-por-autismo-espana-cuenta-ya-con-2116>